TESIS – SS142501

ENTROPY BASED FUZZY SUPPORT VECTOR MACHINE (EFSVM) UNTUK KLASIFIKASI MICROARRAY IMBALANCED DATA

Faroh Ladayya
NRP. 06211550010202

DOSEN PEMBIMBING
Santi Wulan Purnami, M.Si, Ph. D
Irhamah, M.Si, Ph. D

PROGRAM MAGISTER
DEPARTEMEN STATISTIKA
FAKULTAS MATEMATIKA, KOMPUTASI, DAN SAINS DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018

(Halaman ini sengaja dikosongkan)

THESIS - SS142501

ENTROPY BASED FUZZY SUPPORT VECTOR MACHINE (EFSVM) FOR MICROARRAY IMBALANCED DATA CLASSIFICATION

Faroh Ladayya
NRP. 06211550010202

SUPERVISORS
Santi Wulan Purnami, M.Si, Ph. D
Irhamah, M.Si, Ph. D

MAGISTER PROGRAMME
DEPARTEMENT OF STATISTICS
FACULTY OF MATHEMATICS, COMPUTING, AND DATA SCIENCES
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018

ENTROPY BASED FUZZY SUPPORT VECTOR MACHINE (EFSVM) UNTUK KLASIFIKASI MICROARRAY IMBALANCED DATA

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Sains (M.Si)
di
Institut Teknologi Sepuluh Nopember
Oleh:

FAROH LADAYYA
NRP: 06211550010202

Tanggal Ujian : 16 Januari 2018
Periode Wisuda : Maret 2018

Disetujui oleh

1. Santi Wulan Purnami, M.Si., Ph.D (Pembimbing I)
   NIP. 19720923 199803 2 001

2. Irhamah, M.Si., Ph.D (Pembimbing II)
   NIP. 19780406 200112 2 002

3. Dr. Sutikno, M.Si. (Penguji I)
   NIP. 19710313 199702 1 001

4. Dr. rer. pol. Heri Kuswanto, M.Si. (Penguji II)
   NIP. 19820326 200312 1 004

Dekan
Fakultas Matematika, Komputasi, dan Sains Data
Institut Teknologi Sepuluh Nopember

Prof. Dr. Basuki Widodo, M.Sc
NIP. 19650506 198903 1002

(Halaman ini sengaja dikosongkan)

ENTROPY BASED FUZZY SUPPORT VECTOR MACHINE (EFSVM) UNTUK KLASIFIKASI MICROARRAY IMBALANCED DATA

Nama Mahasiswa	**: Faroh Ladayya**
NRP	**: 06211550010202**
Pembimbing	**: Santi Wulan, M.Si, Ph.D**
Co-Pembimbing	**: Irhamah, M.Si, Ph.D**

ABSTRAK

DNA *microarray* merupakan data yang mengandung ekspresi gen dengan ukuran sampel kecil, namun memiliki jumlah *feature* yang sangat besar. Selain itu masalah kelas *imbalanced* merupakan masalah umum dalam data *microarray.* Oleh karena itu diperlukan metode klasifikasi yang mampu mengatasi pemasalahan *high dimensional* dan juga permasalahan *imbalanced*. Metode SVM telah banyak diterapkan untuk klasifikasi data DNA *microarray* dan didapatkan hasil bahwaSVM memberikan kinerja terbaik di antara metode *machine learning* lainnya. Namun pengaruh dari *imbalanced* data pada SVM akan menjadi kekurangan dikarenakan SVM memperlakukan semua sampel dengan kepentingan yang sama sehingga mengakibatkan bias terhadap kelas minoritas. Salah satu metode yang mampu mengatasi *imbalanced* data adalah EFSVM. EFSVM mampu menghasilkan nilai AUC yang tertinggi apabila dibandingkan dengan SVM dan FSVM. Mengingat data DNA *microarray* merupakan *high dimensional data* dengan jumlah *feature* yang sangat besar, maka perlu dilakukan *feature selection* terlebih dahulu. Pada penelitian dilakukan klasifikasi terhadap data DNA *microarray* dengan kasus data yang *imbalanced* menggunakan EFSVM dengan terlebih dahulu dilakukan seleksi fitur menggunakan FCBF. Hasil performansi klasifikasi menunjukkan bahwa *feature selection* mampu meningkatkan performansi klasifikasi. Adanya penambahan *entropy based fuzzy membership* terbukti mampu menghasilkan performansi paling tinggi dibandingkan dengan SVM dan FSVM, namun untuk data yang telah dilakukan *feature selection*, antara FSVM dan EFSVM diperoleh hasil yang hampir sama.

Kata Kunci: DNA Microarray, EFSVM, *High Dimensional*, *Imbalanced Data*

(Halaman ini sengaja dikosongkan)

ENTROPY BASED FUZZY SUPPORT VECTOR MACHINE (EFSVM) FOR MICROARRAY IMBALANCED DATA CLASSIFICATION

Name of Student	**: Faroh Ladayya**
NRP	**: 06211550010202**
Suprvisor	**: Santi Wulan, M.Si, Ph.D**
Co. Supervisor	**: Irhamah, M.Si, Ph.D**

ABSTRACT

DNA microarrays are data containing gene expression with small sample sizes and high number of features. Furthermore, imbalanced classes is a common problem in microarray data. This occurs when a dataset is dominated by a major class which have significantly more instances than the other minority classes in the data. Therefore, it is needed a classification method that can solve the problem of high dimensional and imbalanced data. SVM is one of the classification methods that is capable of handling large or small samples, nonlinear, high dimensional, over learning and local minimum issues. SVM has been widely applied to DNA microarray data classification and it has been shown that SVM provides the best performance among other machine learning methods. However, imbalanced data will be a problem because SVM treats all samples inthe same importancethus the results is bias for minority class. To overcome the imbalanced data, EFSVM is proposed. This method apply a fuzzy membership to each input point and reformulate the SVM such that different input points provide different constributions to the classifier. The samples with higher class certainty, that measured by entropy, are assigned to larger fuzzy membership. The importance of the minority classes have large fuzzy membership and EFSVM can pay more attention to the samples with larger fuzzy membership.Given DNA microarray data is high dimensional data with a very large number of features, it is necessary to do feature selection first using FCBF. Based on the overall results, EFSVM has the highest AUC value compared to SVM and FSVM.

Keywords: DNA Microarray, EFSVM, High Dimensional, Imbalanced Data

(Halaman ini sengaja dikosongkan)

KATA PENGANTAR

Assalamu'alaikumWr. Wb.

Puji syukur kepada Allah S.W.T., atas rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan penyususnan Tesis dengan judul **"ENTROPY BASED FUZZY SUPPORT VECTOR MACHINE UNTUK KLASIFIKASI MICROARRAY IMBALANCED DATA"**. Selain itu tidak lupa sholawat serta salam penulis sampaikan kepada Nabi Muhammad SAW. Dalam menyusun Tesis ini, penulis ucapkan terimakasih kepada pihak-pihak yang membantu dalam menyelesaikan Tesis ini, khususnya kepada :

1. Bapak Dr. Suhartono, M.Sc selaku Ketua Jurusan Statistika FMKSD ITS Surabaya,
2. Bapak Dr.rer.pol. Heri Kuswanto, M.Si selaku Ketua Program Studi Magister Jurusan Statistika ITS Surabaya yang telah memberikan kemudahan dan motivasi kepada semua mahasiswa.
3. Ibu Santi Wulan Purnami, M.Si., Ph.D selaku dosen pembimbing yang telah banyak memberikan arahan, bimbingan, ilmu dan saran serta banyak hal baru yang telah diberikan kepada penulis dalam penyusunan Tesis ini.
4. Ibu Irhamah, M.Si., Ph.D selaku dosen co-pembimbing yang telah banyak memberikan arahan, bimbingan, ilmu dan motivasi kepada penulis dalam penyusunan Tesis ini.
5. Bapak Dr. Sutikno, M.Si selaku dosen penguji yang telah memberikan banyak kritik, saran dan arahan.
6. Bapak Dr.rer.pol. Heri Kuswanto, M.Si selaku dosen penguji sekaligus dosen wali di Program Studi Magister Jurusan Statistika ITS Surabaya.
7. Bapak dan Ibu dosen pengajar di Program Studi Magister Jurusan Statistika ITS Surabaya yang telah memberikan banyak ilmu selama perkuliahan di Program Studi Magister Jurusan Statistika ITS Surabaya

8. Suami, Moh. Nashrulloh, kedua orang tua, Bapak Heri Mujianto dan Ibu Nurul Hidayati, serta Adik, Nanda Roudia Maulin yang telah menjadi penyejuk hati dan tanpa henti memberkan dukungan dan doa kepada penulis.
9. Lovely best friend, Bernadeta, Chusnul, Cordova, dan Kartika atas segalanya.
10. Teman-teman angkatan S2 angkatan genap 2015/2016 yang telah berbagi suka duka selama perkuliahan di Program Studi Magister Jurusan Statistika ITS Surabaya
11. Semua pihak yang tidak dapat disebutkan satu-persatu yang telah membantu hingga Tesis ini dapat terselesaikan dengan baik.

Penulis menyadari sepenuhnya bahwa Tesis ini masih jauh dari sempurna, oleh karena itu segala kritik dan saran yang sifatnya membangun selalu penulis harapkan. Semoga Tesis ini dapat bermanfaat bagi penulis khususnya dan bagi semua yang membutuhkan umumnya. Akhir kata, semoga Allah SWT selalu melimpahkan rahmat serta hidayah-Nya kepada kita semua, Amin amin ya robbal ‘alamiin.

Surabaya, Januari 2018

Penulis

DAFTAR ISI

Halaman

DAFTAR TABEL

(Halaman ini sengaja dikosongkan)

DAFTAR GAMBAR

DAFTAR LAMPIRAN

(Halaman ini sengaja dikosongkan)

DAFTAR NOTASI

H	: Nilai *entropy*
IG	: *Information gain*
SU	: *Symmetrical Uncertainty*
x_i	: *Feature* dari data ke-i
y_i	: Kelas dari data ke-i
$\mathbf{w}$	: Vektor bobot
b	: Variabel bias
d	: Jarak support vektor dengan hyperplane (margin)
L_P	: Lagrangian *primal problem*
L_D	: Lagrangian *dual problem*
α, μ	: Pengali lagrange
ξ_i	: Variabel *slack* pada kasus *linearly non-separable*
C	: Parameter *cost* margin
φ	: Fungsi *mapping non-linear*
$K(\mathbf{x}_i,\mathbf{x}_j)$	: Fungsi Kernel
γ	: Parameter RBF kernel
s	: Fuzzy membership
num_{+i}	: Jumlah anggota kelas positif pada *k nearest neigbors*
num_{-i}	: Jumlah anggota kelas negatif pada *k nearest neigbors*
p_{+i}	: Peluang training data anggota kelas positif pada *k nearest neigbors*
p_{-i}	: Peluang training data anggota kelas negatif pada *k nearest neigbors*
Sub_l	: Subset ke-*l* pada *entropy based fuzzy membership*
H_{+i}	: Nilai *entropy* dari sampel positif ke-*i*
H_{-i}	: Nilai *entropy* dari sampel negatif ke-*i*
H_{max}	: Nilai *entropy* maksimum
H_{min}	: Nilai *entropy* minimum
thrUp	: Nilai batas atas pada tiap subset
thrLow	: Nilai batas bawah pada tiap subset
H_{Sub_l}	: Nilai *entropy* pada subset ke-*l*

β : Parameter *entropy based fuzzy membership*

FM_l : Fuzzy membership pada subset ke-l

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Diagnosis kanker dapat dilakukan berdasarkan struktur morfologisnya, namun menemui kesulitan karena perbedaan struktur morfologis yang sangat tipis antar jenis kanker yang berbeda (Golub, et al., 1999). Kesulitan dalam diagnosis disebabkan pula oleh ketidaklengkapan informasi klinis dari seorang pasien serta kemungkinan adanya faktor subjektivitas pada interpretasi data informasi klinis. Sejumlah kesulitan tersebut mendorong beberapa penelitian dalam diagnosis jenis kanker berdasarkan tingkat ekspresi gen (Ramaswamy, et al., 2001).Teknologi DNA *microarray* menarik minat yang luar biasa pada komunitas ilmiah dan industri dengan kemampuannya untuk mengukur secara simultan aktifitas dan interaksi dari ribuan gen. Teknologi moderen ini menjanjikan wawasan baru dalam mekanisme sistem hidup (Berrar, Dubitzky, & Granzow, 2003). Tipe data ini digunakan untuk mengumpulkan informasi dari jaringan dan sel sampel sehubungan dengan perbedaan ekspresi gen yang dapat dimanfaatkan untuk diagnosa penyakit atau untuk membedakan tipe spesifik dari tumor. Meskipun biasanya ukuran sampel sangat kecil (seringkali kurang dari 100 pasien) untuk training dan testing, namun rentang jumlah *feature* pada data mulai dari 6000 hingga 60.000 (Canedo, Marono, Betanzos, Benitez, & Herrera, 2014). Oleh karena itu data *microarray* merupakan tantangan untuk peneliti sehingga penting untuk menemukan metode yang *robust* (Saeys, Inza, & Larranaga, 2007).

Klasifikasi adalah proses menemukan sekumpulan model atau fungsi yang menggambarkan dan membedakan konsep atau kelas-kelas data dengan tujuan agar model tersebut dapat digunakan untuk memprediksi kelas dari suatu objek atau data yang kelasnya tidak diketahui (Han & Kamber, 2001). Pada umumnya metode klasifikasi yang digunakan adalah analisis diskriminan dan regres ilogistik. Regresi logistik menghasilkan nilai yang berupa probabilitas sehingga kurang praktis (Yohannes & Webb, 1999). Sedangkan untuk analisis diskriminan terikat beberapa asumsi yaitu variable prediktor harus berskala rasio atau interval,

matriks varian yang sama untuk setiap populasi dan data harus berdistribusi normal multivariate (Brieman, Friedman, Olshen, & C, 1984). Karena metode tersebut memiliki kelemahan sehingga muncul metode *machine learning* yang dikembangkan untuk membantu klasifikasi tanpa terikat oleh asumsi dan memberikan fleksibilitas analisis data yang lebih besar tetapi tetap menghasilkan tingkat akurasi yang tinggi dan mudah penggunaannya, antara lain, *Multivariate Adaptive Regression Spilne* (MARS), *Feed-Forward Neural Network* (FFNN), *K-Nearest Neighbours* (K-NN*), Classification Adaptive Regression Tree* (CART), *Artificial Neural Network*, dan *Support VectorMachine* (SVM) (Scholkopf & Smola, 2002).

SVM adalah metode klasifikasi yang didasarkan pada teori *statisticallearning*. Dalam berbagai aplikasi SVM menghasilkan hasil klasifikasi yang lebih baik daripada metode parametrik (misalnya analisis logit) dan mengalahkan teknik nonparametrik lainnya yang digunakan secara luas, seperti jaringan saraf (Hardle & Simar, 2015). Menurut teori *Structural Risk Minimization* (SRM), SVM telah memperlihatkan perfoma sebagai metode yang bisa mengatasi masalah *overfitting* dengan cara meminimalkan batas atas pada *generalization error* yang menjadi alat yang kuat untuk *supervisedlearning*. SVM dapat menangani sampel besar atau kecil, non-linear, high dimensional, *over learning* dan masalah lokal minimum (Guo, Yi, Wang, Ye, & Zhao, 2014). Menurut Rachman(2012), Huang (2007) dan Byvatov (2003) Support Vector Machine memiliki tingkat akurasi yang lebih baik jika dibandingkan dengan metode regresi logistik, ANN, Naive Bayes, dan CART. Lee dan To (2010) membandingkan SVM dan *Back Propagation Neural Network* untuk mengevaluasi krisis keuangan perusahaan, didapatkan hasil bahwa SVM lebih baik dibandingkan dengan *Back Propagation Neural Network*.

Selain penelitian yang telah disebutkan sebelumnya, metode SVM juga telah banyak diterapkan untuk klasifikasi data DNA *microarray*. Guyon (2002) menggunakan metode SVM berbasis *RecursiveFeatureElimination* (RFE) untuk membuat *classifier*yang cocok pada diagnosis genetik, serta penemuan obat menggunakan data DNA *microarray*. Chu (2005) menggunakan SVM untuk klasifikasi kanker dengan data *microarray*. Lee (2004) menilai kinerja beberapa

metode melalui studi banding yang komprehensif. Penelitian ini menunjukkan bahwa pengklasifikasi lebih canggih memberikan kinerja yang lebih baik daripada metode klasik seperti *k-Nearest Network* (kNN), DLDA, DQDA dan pilihan metode seleksi gen memiliki banyak berpengaruh pada kinerja metode klasifikasi, SVM memberikan kinerja terbaik di antara metode *machinelearning* lainnya di sebagian besar dataset terlepas dari pemilihan gen. Seeja & Shweta (2011) mengklasifikasikan ekspresi gen pada DNA microarray menggunakan SVM dan didapatkan hasil bahwa SVM lebih baik daripada *Neural Network*.

Imbalanced data menjadi salah satu tantangan pada komunitas data mining(Sun, Wong, & Kamel, 2009). Kelas *imbalanced* terjadi ketika sebuah data didominasi oleh kelas mayoritas atau kelas yang secara signifikan lebih banyak kejadian daripada kelas yang langka atau kelas minoritas (Canedo, Marono, Betanzos, Benitez, & Herrera, 2014). Umumnya kelas dengan jumlah yang banyak ditandai sebagai kelas negatif. Sedangkan kelas dengan jumlah sampel yang sedikit dinamakan kelas positif. Fan, dkk (2016) dalam penelitiannya membagi *Imbalanced Ratio* (IR), dimana *IR* merupakan hasil bagi antara jumlah kelas negatif (mayoritas) dengan kelas positif (minoritas), kedalam tiga kelompok yaitu *Low Imbalanced* (IR $\leq$ 9,0), *Medium Imbalanced* (9.0 < *IR* $\leq$ 20), dan *High Imbalanced* (IR > 20). Masalah kelas *imbalanced* merupakan masalah umum dalam data *microarray*(Canedo, Marono, Betanzos, Benitez, & Herrera, 2014). Misalnya, kelas kanker cenderung jarang daripada kelas non-kanker karena biasanya ada pasien lebih sehat. Namun, penting bagi praktisi untuk memprediksi dan mencegah munculnya kanker. Dalam kasus ini, algoritma klasifikasi standar memiliki bias terhadap kelas dengan jumlah kasus yang lebih besar, kejadian kelas minoritas lebih sering terjadi kesalahan klasifikasi daripada kejadian dari kelas-kelas lain (Galar, Fernandez, Barrenchea, Bustince, & Herrera, 2012).Namun sebagian besar algoritma klasifikasi standar ditujukan untuk distribusi kelas yang *balance* atau biaya kesalahan klasifikasi yang sama (Brown & Mues, 2012).

Berdasarkan beberapa metode klasifikasi yang telah disebutkan sebelumnya, SVM memiliki kemampuan yang lebih baik dalam klasifikasi daripada metode lain. Namun pengaruh dari *imbalanced* data pada SVM akan

menjadi kekurangan dalam paradigma memaksimalkan margin (Akbani, Kwek, &Japkowicz, 2004). Karena SVM memperlakukan semua sampel dengan kepentingan yang sama dan mengabaikan antara kelas positif dan negatif yang mengakibatkan daerah keputusan menjadi bias terhadap kelas yang negatif (Fan, Wang, Li, Gao, & Zha, 2016). Untuk menangani masalah *imbalanced*, beberapa penelitian telah dilakukan. Penelitian tersebut dapat dibagi kedalam tiga kelompok yaitu, pendekatan pada tingkat data (Galar, Fernandez, Barrenechea, & Herrera, 2013), pendekatan pada tingkat algoritma (Maldonado & Lopez, 2014) dan kombinasi antara pendekatan data dan algoritma (Chawla, Cieslak, Hall, & Joshi, 2008). Pendekatan pada tingkat data dibagi menjadi tiga yaitu, metode *over-sampling*, *under-sampling*, dan metode*hybrid*. Salah satu metode *over-sampling* yang terkenal adalah *Synthetic Minority Over sampling Technique* (SMOTE) yang penelitiannya telah dilakukan olehBowyer, Chawla, Hall, & Kegelmeyer (2002) dan Trapsilasiwi (2013). Metode *under-sampling* mencoba untuk menyeimbang-kan data dengan menghapus beberapa sampel dari kelas negatif. Salah satu metode under-sampling yang terkenal adalah *One-Side Selection* (OSS) (Kubat & Matwin, 1997). Selanjutnya penelitian menggunakan metode*hybrid* yang merupakan kombinasi antara *over-sampling* dan *under-sampling* telah dilakukan oleh Batista, Ronaldo, & Monard (2004) dan Khaulasari (2016).Sedangkan untuk pendekatan pada tingkat algoritma, Liu, Wu, dan Zhou (2009) mengusulkan *Easy Ensemble*. *Easy ensemble* mengambil sampel beberapa subset dari kelas negatif dan membentuk *classifier* dari masing-masing sampel setelah itu, *output* dari *classifier* tersebut dikombinasikan.

Selain beberapa metode tersebut, Lin dan Wang (2002) mengusulkan *Fuzzy* SVM (FSVM) yang berlaku keanggotaan *fuzzy* untuk masing-masing sampel dan merumuskan SVM, sehingga input sampel yang berbeda memiliki kontribusi yang berbeda untuk *learningdecision surface*. Untuk menentukan fungsi keanggotaan *fuzzy* adalah titik kunci di FSVM. Meskipun mudah digunakan, FSVM masih memiliki kelemahan dalam menangani data *imbalanced*, misalnya akurasi kesalahan klasifikasi kelas positif lebih tinggi dari kelas negatif. Lin dan Wang (2002) dalam penelitiannya juga mengusulkan untuk diterapkan

fungsi keanggotaan *fuzzy* yang tepat agar dapat secara otomatis menentukan model keanggotaan *fuzzy*.

Dalam rangka untuk mengatasi kelemahan dari FSVM, Tian, Peng dan Ha (2012) memperkenalkan sebuah novel keanggotaan *fuzzy* yang menentukan fungsi dan mengusulkan FSVM baru berdasarkan data non-ekuilibrium yang secara efektif mengurangi akurasi kesalahan klasifikasi kelas yang positif. Batuwita dan Palade (2010) mengusulkan metode untuk meningkatkan FSVM untuk kelas *imbalanced*, yang digunakan untuk menangani masalah kelas *imbalanced* terhadap adanya *outlier* dan *noise*. Wang dkk. (2005) mengusulkan *Bilateral-weighted* FSVM (B-FSVM) pada credit scoring area, B-FSVM memperlakukan setiap sampel sebagai kelas positif dan negatif tetapi memiliki keanggotaan *fuzzy* yang berbeda. Fan dkk. (2016) mengusulkan evaluasi keanggotaan *fuzzy* baru yang menetapkan keanggotaan *fuzzy* setiap sampel berdasarkan kepastian kelasnya menggunakan *entropy*.

Dalam teori informasi, *entropy* adalah ukuran efektif untuk kepastian. Shannon (2001) mendefinisikan *entropy* sebagai fungsi logaritmik negatif probabilitas terjadinya suatu peristiwa. Dalam penelitiannya,kepastian kelas menunjukkan kepastian dari sampel yang diklasifikasikan ke kelas tertentu. Karena *entropy* adalah sebuah pendekatan yang efektif dalam mengukur kepastian, maka *entropy* digunakan untuk mengevaluasi kepastian kelas masing-masing sampel. Dengan evaluasi keanggotaan *fuzzy* berbasis *entropy*, FSVM berbasis *Entropy* (EFSVM) memiliki kapasitas yang kuat berhadapan dengan data *imbalanced* (Fan, Wang, Li, Gao, & Zha, 2016). Dalam prakteknya, kepentingan kelas positif lebih tinggi dari yang negatif dalam fenomena *imbalanced*, *classifier* harus lebih memperhatikan sampel positif daripada yang negatif. Dengan demikian, sampel positif ditetapkan untuk keanggotaan *fuzzy* relatif lebih besar besar agar menjamin kepentingannya. Sementara, keanggotaan *fuzzy* sampel negatif ditentukan dengan metode evaluasi keanggotaan *fuzzy* berdasarkan *entropy*, sehingga kepentingan kelas negatif harus melemah dalam *learning process*.

Pada penelitiannya Fan dkk. (2016) membandingkan EFSVM dengan SVM dan FSVM, hasilnya menujukkan bahwa EFSVM memiliki nilai persentase AUC

(*Area Under Curve*) tertinggi diantara kedua metode tersebut, selain itu Fan (2016) juga membandingkan EFSVM dengan beberapa metode SVM dengan pendekatan yang lain seperti, SVM-SMOTE, SVM-OSS, SVM-RUS, dan didapatkan hasil bahwa EFSVM memiliki rata-rata AUC yang tertinggi baik untuk data dengan *imbalanced* ratio yang rendah, medium, dan tinggi. Melihat kelebihan dari EFSVM dibandingkan dengan SVM, dan beberapa SVM dengan pendekatan lain, maka pada penelitian ini akan dilakukan klasifikasi data DNA *microarray* menggunakan EFSVM.

Mengingat data DNA *microarray*merupakan *high dimensional data* dengan jumlah *feature* yang sangat besar, maka perlu dilakukan *feature selection* terlebih dahulu saat *preprocessing data*. *Feature* selection adalah salah satu proses dalam *prepocessing* data yang digunakan untuk memilih *feature* atau variabel terbaik. *Feature* selection merupakan bagian penting untuk mengoptimalkan kinerja dari *classifier* (Wang, Li, Song, Wei, & Li, 2011). Beberapa metode filter *feature* selection antara lain information gain (IG), Chi-Square, *Correlation-based feature selection* (CFS), *Fast Correlation Based Filter* (FCBF) dan *Consistency Based Filter* (CBF) (Alonso, Noelia, & Veronica, 2015). FCBF merupakan algoritma *feature* selection baru yang terbukti bekerja secara cepat dan mampu memilih fitur yang terbaik serta mempertimbangkan kecepatan waktu (Liu & Lei, 2003). Sebelum dilakukan analisis menggunakan EFSVM terlebih dahulu akan dilakukan seleksi *feature* pada data DNA *microarray* menggunakan FCBF.

Pada penelitian akan dilakukan klasifikasi terhadap data DNA *microarray* dengan kasus data yang *imbalanced* menggunakan EFSVM. Sebelum data diklasifikasi menggunakan EFSVM terlebih dahulu dilakukan seleksi fitur menggunakan FCBF. Data DNA *microarray* yang akan digunakan adalah data *breast cancer* dan *colon cancer* dengan nilai *IR* 1,9 untuk data *breast cancer* dan 1,8 untuk data kanker *colon cancer*. Hasil performansi klasifikasinya akan dinyatakan dengan AUC, akurasi, *sensitivity*, *specificity*, dan *g-means* kemudian nilainya dibandingkan dengan SVM dan FSVM.

1.2 Rumusan Masalah

Data DNA *microarray* merupakan *high dimensional* data dengan jumlah *feature* yang sangat besar namun jumlah sampel sangat terbatas. Hal ini tentunya menjadi tantangan dalam klasifikasi terlebih lagi data DNA *microarray* umumnya memiliki kondisi yang *imbalanced.* Akibatnya dihasilkan akurasi prediksi yang baik terhadap kelas negatif (mayoritas) sedangkan untuk kelas positif (minoritas) akan dihasilkan akurasi prediksi yang buruk. Dari uraian diatas maka permasalahan dalam penelitian ini adalah mengkaji dan menerapkan *Entropy based Fuzzy Support Vector Machine* (EFSVM) untuk klasifikasi *imbalanced data* dan menggunakan FCBF sebagai *feature selection*.

1.3 Tujuan Penelitian

Berdasarkan permasalahan, tujuan dari penelitian ini adalah sebagai berikut.

1. Mengkaji metode *Entropy based Fuzzy Support VectorMachine* (EFSVM)
2. Menerapkan EFSVM pada data *imbalanced* DNA *microarray*
3. Mengetahui hasil performansi dari metode EFSVM dibandingkan dengan SVM dan FSVM berdasarkan nilai AUC, akurasi, *sensitivity, specificity*, dan *g-means.*

1.4 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah

1. Memberikan alternatif metode klasifikasi khususnya untuk data *high dimensional* dan *imbalanced*
2. Menambah wawasan keilmuan Statistika utamanya dibidang *data mining*.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut.

1. Feature selection yang digunakan adalah *Fast Correlation Based Filter* (FCBF)
2. Kernel yang digunakan adalah RBF Kernel
3. Klasifikasi yang dilakukan adalah *binary classification*.

(Halaman ini sengaja dikosongkan)

BAB 2
TINJAUAN PUSTAKA

Pada bab ini dibahas mengenai metode yang digunakan dalam penelitian ini yaitu, *Fast Correlation Based Filter* (FCBF), Support Vector Machine (SVM), *Entropy based Fuzzy* SVM (EFSVM), dan evaluasi performansi metode klasifikasi.

2.1 *Fast Correlation Based Filter* (FCBF)

FCBF merupakan salah satu algoritma *feature selection* yang bersifat multivariate dan mengukur kelas *feature* dan korelasi antara *feature-feature* (Alonso, Noelia, & Veronica, 2015). Secara umum, *feature* dikatakan bagus jika *feature* tersebut relevan dengan konsep kelas namun tidak redundan pada *feature* yang lain. Jika diterapkan korelasi antara dua variabel sebagai ukuran kebaikan, maka sebuah *feature* dikatakan bagus untuk klasifikasi jika berkorelasi sangat tinggi dengan kelas namun tidak berkorelasi dengan *feature* lainnya. Namun pengukuran dengan korelasi tidak mampu menangkap korelasi yang tidak *linear* selain itu korelasi mengharuskan semua feature dan kelas mengandung nilai numerik. Untuk mengatasi kekurangan ini, Yu dan Liu (2009), menerapkan pendekatan lain yaitu memilih ukuran korelasi berdasarkan konsep *information-theoretical entropy*. *Entropy* dari variabel X didefinisikan pada Persamaan (2.1).

$$H(X) = -\sum_{i}^{n} P(x_i)\log_2(P(x_i)),\ i = 1,2,...,\mathrm{n} \tag{2.1}$$

Entropy dari variabel X jika diketahui variabel Y didefinisikan pada Persamaan (2.2).

$$H(X \mid Y) = -\sum_{i}^{n} P(\mathrm{y}_i)\sum_{i}^{n} P(x_i \mid \mathrm{y}_i)\log_2(P(x_i \mid \mathrm{y}_i)),\ i = 1,2,...,\mathrm{n} \tag{2.2}$$

$P(x_i)$ adalah posterior probabilities untuk semua nilai X dan $P(x_i|y_i)$ adalah posterior probabilities dari X jika Y diketahui. Dari *entropy* tersebut dapat diperoleh *Information Gain* sebagai berikut:

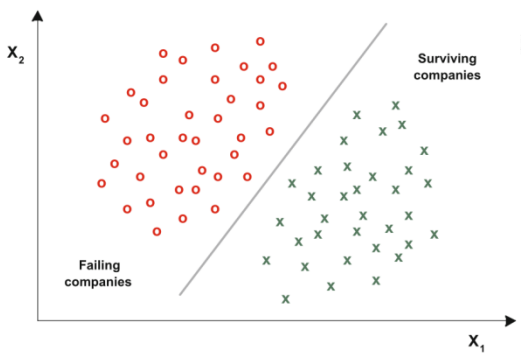
$$IG(X \mid Y) = H(X) - H(X \mid Y) \tag{2.3}$$

Untuk mengukur korelasi antar *feature*, maka digunakan *symmetrical uncertainty*. Nilai *symmetrical uncertainty* berkisar pada rentang 0 sampai dengan 1. *Symmetrical uncertainty* dirumuskan sebagai berikut.

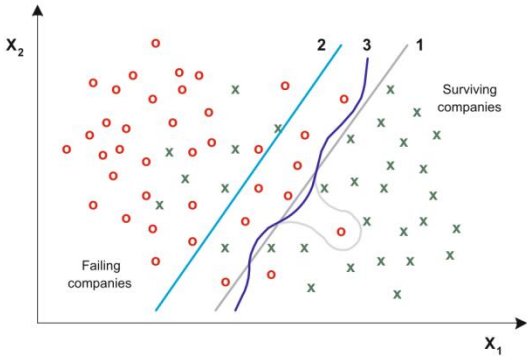
$$SU(X,Y) = 2\frac{IG(X \mid Y)}{H(X) + H(Y)} \tag{2.4}$$

2.2 *Support Vector Machine*

Dasar *Support Vector Machine* (SVM) telah dikembangkan oleh Vapnik (1995) dan mendapatkan popularitas karena banyak *feature* yang menarik, dan kinerja empiris menjanjikan. SVM didasarkan pada prinsip *Structural Risk Minimization* (SRM). Formulasi mewujudkan prinsip *Structural Risk Minimization* (SRM), telah terbukti lebih unggul (Gunn, 1998). Prinsip induksi ini berbeda dengan prinsip minimalisasi resiko empirik yang hanya meminimalkan kesalahan proses training. Pada SVM fungsi tujuan dirumuskan sebagai masalah optimisasi konveks berbasis *quadratic programming*, untuk menyelesaikan dual problem. SVM adalah metodologi yang kuat untuk menyelesaikan masalah non*linear* dan fungsi klasifikasi yang menyebabkan banyak perkembangan baru lainnya dalam metode berbasis kernel. Dalam klasifikasi menggunakan SVM, peneliti diharuskan untuk menemukan fungsi pemisah antar kelas. Margin (m) adalah jarak antara *hyperplane* dengan data terdekat pada masing-masing kelas. Bidang pembatas pertama membatasi kelas pertama dan bidang pembatas kedua membatasi kelas kedua sedangkan data yang berada pada bidang pembatas merupakan vektor-vektor yang terdekat dengan *hyperplane* terbaik disebut dengan *Support Vector* (Tan, Steinbach, & Kumar, 2006). SVM untuk klasifikasi dapat bekerja pada kasus klasifikasi *linear* maupun non*linear*, seperti diilustrasikan pada Gambar 2.1. Pada klasifikasi *linear*, SVM dapat dibedakan menjadi dua yaitu *linearly separable* dan *linearly non-separable*.



Gambar 2.1 (a) Fungsi Klasifikasi *Linear* pada Kasus *Linearly Separable* (Diperoleh dariHardle & Simar, 2015)



Gambar 2.1 (b) Fungsi Klasifikasi *linear* yang berbeda (1) dan (2) dan fungsi non *linear* (3) pada kasus linearly non-separable (Diperoleh dari Hardle & Simar, 2015)

2.2.1 SVM pada Kasus *Linearly Separable*

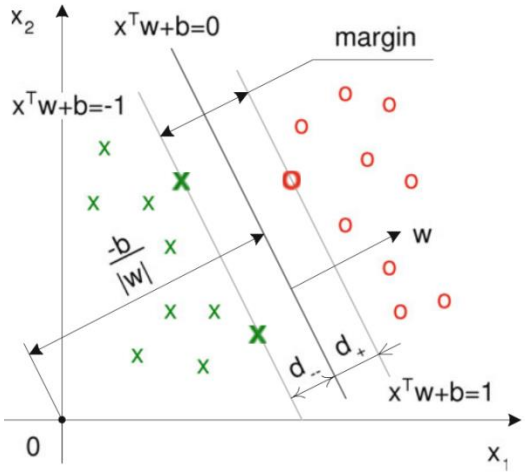
Terlebih dahulu akan didiskripsikan SVM pada kasus *linearly separable*. Keluarga $\mathcal{F}$ fungsi klasifikasi pada ruang data diberikan oleh:

$$\mathbf{F} = \left\{ \mathbf{x}'\mathbf{w} + b, \mathbf{w} \in \mathbf{R}^p, b \in \mathbf{R} \right\} \tag{2.5}$$

Untuk menentukan *Support Vector* dipilih $f \in \mathbf{F}$ (atau ekuivalen (**w,b**)) seperti yang disebut margin, koridor antara *hyperplane* yang memisahkan, maksimal. Situasi tersebut ditunjukkan pada Gambar 2.2. Margin sama dengan $d_- + d_+$. Fungsi klasifikasi adalah *hyperplane* ditambah zona margin. Ini memisahkan poin dari kedua kelas dengan jarak "paling aman" tertinggi (margin) diantara mereka. Hal ini dapat ditunjukkan bahwa maksimalisasi margin sesuai dengan pengurangan kompleksitas. Pemisahan *hyperplane* didefinisikan hanya dengan *Support Vector* yang menahan *hyperplane* sejajar dengan pemisah.

$\mathbf{x}_i'\mathbf{w} + b = 0$ adalah *hyperplane* pemisah, $d_- + d_+$ akan menjadi jarak terpendek pada objek yang paling dekat dari kelas +1 (-1). Karena pemisahan dapat diselesaikan tanpa error, semua observasi *i = 1, 2, ..., n* harus memenuhi:

$$\mathbf{x}_i'\mathbf{w} + b \geq +1 \text{ untuk } y_i = +1$$
$$\mathbf{x}_i'\mathbf{w} + b \geq -1 \text{ untuk } y_i = -1$$



Gambar 2.2 Hyperplane Pemisah dan Margin pada Kasus Linearly Separable (Diperoleh dari Hardle & Simar, 2015)

kedua konstrain dapat dikombinasikan menjadi satu:

$$y_i(\mathbf{x}_i'\mathbf{w}+b)-1\geq 0 \quad i=1,2,...,n \tag{2.6}$$

canonical *hyperplane*s $\mathbf{x}_i'\mathbf{w}+b=\pm 1$sejajar dan jarak antara masing-masing dan *hyperplane*s pemisah adalah $d_+ = d_- = 1/\|\mathbf{w}\|$dimana w adalah vektor bobot (*weight vector*) yang berukuran (p×1), b adalah posisi bidang relatif terhadap pusat koordinat atau lebih dikenal dengan bias yang bernilai skalar.

$$X=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n1} & x_{n1} & \cdots & x_{np} \end{bmatrix} \quad \mathbf{x}_i'=\begin{bmatrix} x_{i1} & x_{i2} & \cdots & x_{ip} \end{bmatrix} \quad \mathbf{w}=\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{bmatrix} \quad y_i=\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Gambar 2.2 menunjukkan $\frac{|b|}{\|\mathbf{w}\|}$ adalah jarak bidang pemisah yang tegak lurus dari titik pusat koordinat dan $\|\mathbf{w}\|$ adalah jarak *euclidean* (norm *euclidean*) dari **w**. Panjang vektor **w** adalah norm $\|\mathbf{w}\|=\sqrt{\mathbf{w}'\mathbf{w}}=\sqrt{w_1^2+w_2^2+...+w_p^2}$. Bidang batas pertama membatasi kelas (+1) sedangkan bidang pembatas kedua membatasi kelas (-1). Bidang pembatas pertama $\mathbf{x}_i'\mathbf{w}+b=1$ mempunyai bobot **w** dan jarak tegak lurus dari titik asal sebesar $\frac{|1-b|}{\|\mathbf{w}\|}$, sedangkan bidang pembatas kedua

$\mathbf{x}_i'\mathbf{w}+b=-1$mempunyai bobot $\mathbf{w}$ dan jarak tegak lurus dari titik asal sebesar $\frac{|-1-b|}{\|\mathbf{w}\|}$. Nilai maksimum margin atau nilai jarak antar bidang pembatas adalah

$$\frac{1-b-(-1-b)}{\|\mathbf{w}\|}=\frac{2}{\|\mathbf{w}\|} \tag{2.7}$$

Secara matematis, formulasi permasalahan optimasi SVM untuk klasifikasi *linear* dalam primal space adalah

$$\min \frac{1}{2}\|\mathbf{w}\|^2 \tag{2.8}$$

dengan fungsi kendala $y_i(\mathbf{x}_i'\mathbf{w}+b)\geq 1,\ i=1,2,...n$

Pada formulasi diatas, ingin memaksimalkan fungsi tujuan $\frac{1}{2}\|\mathbf{w}\|^2$ atau sama saja dengan memaksimalkan $\|\mathbf{w}\|^2$ atau $\|\mathbf{w}\|$.

Secara umum persoalan optimasi (2.8) akan lebih mudah diselesaikan jika diubah ke dalam formula lagrange. Lagrangian dari primal problem yang berhubungan dengan maksimalisasi margin adalah:

$$L_p(\mathbf{w},\mathrm{b},\alpha)=\frac{1}{2}\|\mathbf{w}\|^2-\sum_{i=1}^{n}\alpha_i\left[y_i(\mathbf{x}_i'\mathbf{w}+b)-1\right] \tag{2.9}$$

Karush-Kuhn-Tucker (KKT) (Gale et al 1951) kondisi optimal order pertama adalah

$$\begin{aligned}
&\frac{\partial L_p(\mathbf{w},\mathrm{b},\alpha)}{\partial \mathbf{w}}=0: \qquad \mathbf{w}-\sum_{i=1}^{n}\alpha_i y_i\mathbf{x}_i=0\\
&\frac{\partial L_p(\mathbf{w},\mathrm{b},\alpha)}{\partial b}=0: \qquad \sum_{i=1}^{n}\alpha_i y_i=0\\
&y_i(\mathbf{x}_i'\mathbf{w}+b)-1\geq 0, \quad i=1,2,...,n\\
&\alpha_i\geq 0\\
&\alpha_i\left\{y_i(\mathbf{x}_i'\mathbf{w}+b)-1\right\}=0
\end{aligned} \tag{2.10}$$

Substitusi hasil KKT persamaan (2.10) pada persamaan (2.9) sehingga didapatkan lagrangian untuk dual problem:

$$L_D(\alpha)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j\mathbf{x}_i'\mathbf{x}_j \tag{2.11}$$

primal dan dual problem adalah:

$$\min_{\mathbf{w},b,\alpha} L_p(\mathbf{w}, \mathrm{b}, \alpha) \tag{2.12}$$

$$\max_{\alpha} L_D(\alpha) \quad s.t. \quad \alpha_i \geq 0, \quad \sum_{i=1}^{n} \alpha_i y_{\ i} = 0 \tag{2.13}$$

Karena optimisasi problem adalah *convex* maka rumusan dual dan primal memberikan solusi yang sama.

Titik-titik i untuk persamaan $y_i(\mathbf{x}_i'\mathbf{w} + b) = 1$disebut *Support Vector*. Setelah "training SVM" yaitu menyelesaikan dual problem diatas dan mendapatkan *lagrange multiplier* (sama dengan 0 untuk non-*Support Vector*s) dapat diklasifikasikan sebagai sebuah kelompok dengan menggunakan aturan klasifikasi:

$$g(x) = sign(\mathbf{x}_i'\mathbf{w} + b) \tag{2.14}$$

dimana $\mathbf{w} = \sum_{i=1}^{n} a_i y_i x_i$ dan $b = \frac{1}{2}\left(x_{+1} + x_{-1}\right) w.x_{+1}$ dan x_{-1} adalah dua *Support Vector* yang mengikuti kelas yang berbeda untuk $y\left(\mathbf{x}'\mathbf{w} + b\right) = 1$. Nilai dari fungsi klasifikasi (skor dari kelompok) dapat dihitung sebagai berikut.

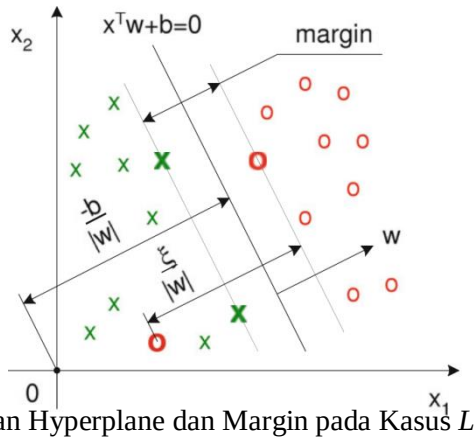
$$f(x) = \mathbf{x}_i'\mathbf{w} + b \tag{2.15}$$

masing-masing skor *f(x)* secara unik berhubungan dengan *default probability* (PD). Semakin tinggi *f(x)* maka semaki tinggi pula PD (Hardle & Simar, 2015).

2.2.2 SVM pada Kasus *Linearly Non-separable*

Pada kasus linearly non-separable situasi ditunjukkan pada Gambar 2.3. Variabel slack ξ_i menunjukkan menunjukkan pinalti terhadap ketelitian pemisahan. Pada kasus ini ketidaksamaan didefinisikan sebagai berikut.

$$\begin{aligned} \mathbf{x}_i'\mathbf{w} + b &\geq 1 - \xi_i \quad \text{for } y_i = 1 \\ \mathbf{x}_i'\mathbf{w} + b &\leq 1 + \xi_i \quad \text{for } y_i = -1 \\ \xi_i &\geq 0 \end{aligned} \tag{2.16}$$



Gambar 2.3 Pemisahan Hyperplane dan Margin pada Kasus *Linearly Non-separable* (Diperoleh dari Hardle & Simar, 2015)

Dapat dikombinasikan kedalam dua konstrain:

$$\begin{aligned} y_i(\mathbf{x}_i'\mathbf{w}+b) &\geq 1-\xi_i \\ \xi_i &\geq 0 \end{aligned} \tag{2.17}$$

Klasifikasi SVM memaksimumkan margin yang diberikan keluarga fungsi klasifikasi $\boldsymbol{\mathcal{F}}$. Penalti dari kesalahan klasifikasi, *classification error* $\xi_i \geq 0$ berhubungan dengan jarak dari titik kesalahan klasifikasi x_i pada *canonical hyperplane* menghubungkan kelasnya. Fungsi tujuan yang sesuai dengan penalti maksimalisasi margin dirumuskan sebagai berikut.

$$\min_{\mathbf{w},\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \tag{2.18}$$

dimana parameter C menggolongkan bobot yang diberikan kepada kesalahan klasifikasi. Minimalisasi fungsi tujuan dengan kendala (2.17) menghasilkan kemungkinan margin tertinggi dari *hyperplane* pemisah. Fungsi *lagrange* untuk primal problem adalah:

$$L_P(\mathbf{w},b,\alpha,\mu,\xi) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i\{y_i(\mathbf{x}_i'\mathbf{w}+b)-1+\xi_i\} - \sum_{i=1}^{n}\mu_i\xi_i \tag{2.19}$$

dimana $\boldsymbol{\alpha_i \geq 0}$ dan $\boldsymbol{\mu_i \geq 0}$ adalah pengali lagrange. Primal problem dirumuskan sebagai berikut.

$$\min_{\mathbf{w},b,\alpha,\mu,\xi} L_P(\mathbf{w},b,\alpha,\mu,\xi) \tag{2.20}$$

Kondisi order pertama pada kasus ini adalah:

$$\begin{aligned}
\frac{\partial L_P(\mathbf{w},b,\alpha,\mu,\xi)}{\partial \mathbf{w}} = 0: \quad & \mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \\
\frac{\partial L_P(\mathbf{w},b,\alpha,\mu,\xi)}{\partial b} = 0: \quad & \sum_{i=1}^{n} \alpha_i y_i = 0 \\
\frac{\partial L_P(\mathbf{w},b,\alpha,\mu,\xi)}{\partial \xi_i} = 0: \quad & \mathrm{C} - \alpha_i - \mu_i = 0
\end{aligned} \tag{2.21}$$

dengan syarat dari pengali lagrange:

$$\begin{aligned}
&\alpha_i \geq 0 \\
&\mu_i \geq 0 \\
&\alpha_i \left\{ y_i(\mathbf{x}_i'\mathbf{w} + b) - 1 + \xi_i \right\} = 0 \\
&\mu_i \xi_i = 0
\end{aligned} \tag{2.22}$$

Perhatikan bahwa $\sum_{i=1}^{n} \alpha_i y_i = 0$ oleh karena itu mirip dengan kasus linear separable, masalah primal diterjemahkan menjadi:

$$\begin{aligned}
L_D(\alpha) = & \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i' \mathbf{x}_j - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i' \sum_{j=1}^{n} \alpha_i y_i \mathbf{x}_j \\
& + C \sum_{i=1}^{n} \xi_i + \sum_{j=1}^{n} \alpha_i - \sum_{j=1}^{n} \alpha_i \xi_i - \sum_{j=1}^{n} \mu_i \xi_i \\
= & \sum_{j=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i' \mathbf{x}_j + \sum_{i=1}^{n} \xi_i (C - \alpha_i - \mu_i)
\end{aligned} \tag{2.23}$$

Karena bentuk terakhir adalah 0 maka diperoleh dual problem sebagai berikut (Hardle & Simar, 2015).

$$L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i' \mathbf{x}_j \tag{2.24}$$

dan dual problem sebagai berikut:

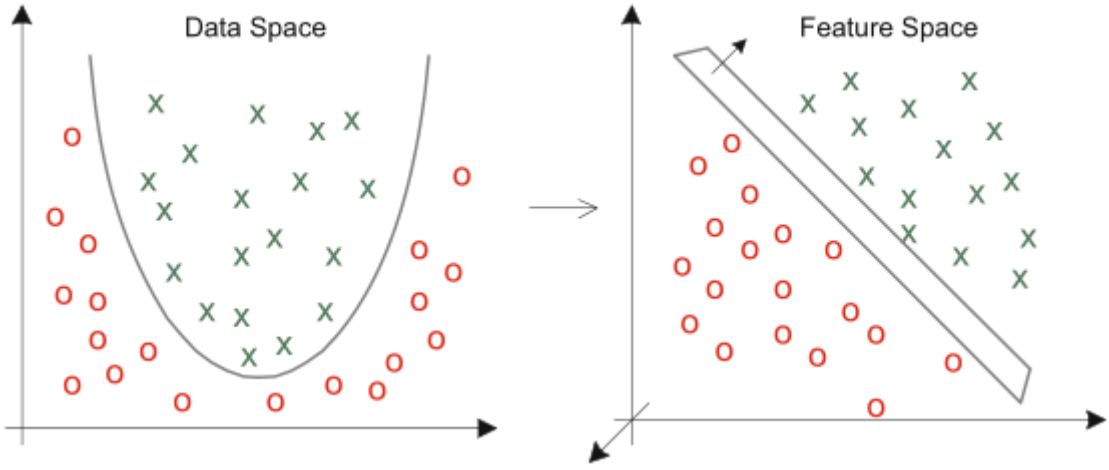
$$\max_{\alpha} L_D(\alpha),$$

dimana,

$$\begin{aligned}
&0 \leq \alpha_i \leq C, \\
&\sum_{i=1}^{n} \alpha_i y_i = 0
\end{aligned} \tag{2.25}$$

2.2.3 Klasifikasi *Non-linear* SVM

SVM dapat digeneralisasi untuk kasus *non-linear*. Untuk mendapatkan *non-linear classifier* seperti yang ditunjukkan pada Gambar 2.4 yang memetakan data dengan struktur non-*linear* melalui fungsi $\varphi: \mathbb{R}^p \longmapsto \mathbb{H}$ kedalam ruang dimensi yang sangat besar, $\mathbb{H}$, dimana aturan klasifikasi (hampir) *linear*. Perhatikan bahwa semua vektor training x_i muncul pada L_D (2.24) hanya sebagai hasil kali pada bentuk $\mathbf{x}_i'\mathbf{x}_j$. Pada SVM *non-linear* hal tersebut ditransformasi kedalam bentuk $\varphi(\mathbf{x}_i')\varphi(\mathbf{x}_j)$.



Gambar 2.4 Fungsi Klasifikasi non-*linear* (Diperoleh dari Hardle & Simar, 2015)

Fungsi transformasi pada SVM adalah menggunakan "Kernel Trick" (Scholkopf & Simola, 2002). Kernel trick adalah menghitung scalar product dalam bentuk sebuah fungsi kernel.

$$K(\mathbf{x}_i,\mathbf{x}_j) = \varphi(\mathbf{x}_i')\varphi(\mathbf{x}_j) \tag{2.26}$$

jika sebuah fungsi kernel K pada Persamaan (2.26) ini dapat digunakan tanpa perlu mengetahui fungsi transformasi φ secara eksplisit. Syarat perlu dan cukup pada fungsi simetrik $K(\mathbf{x}_i,\mathbf{x}_j)$ untuk menjadi kernel diberikan oleh teorema Mercer. Diberikan sebuah kernel K dan data $x_1, x_2, \dots, x_n \in X$ maka matriks $K = \left(K(\mathbf{x}_i, \mathbf{x}_j)\right)_{ij}$ berukuran $n \times n$ disebut *Gram matrix* untuk data $x_1, x_2, \dots, x_n$. Sebuah syarat cukup dan perlu untuk matriks simetri K dengan $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) =$

$K(\mathbf{x}_i, \mathbf{x}_j) = K_{ji}$ untuk K definit positif disebut "*Mercer's Theorem*" (Mercer, 1909)

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\varphi_i\varphi_j K(\mathbf{x}_i,\mathbf{x}_j) \geq 0 \tag{2.27}$$

Contoh sederhana pada sebuah kernel trick yang menunjukkan bahwa kernel dapat dihitung tanpa perhitungan fungsi *mapping* φ secara eksplisit adalah fungsi pemetaan:

$$\varphi(x_1, x_2) = (x_1^2, \sqrt{2}x_1x_2, x_2^2)' \tag{2.28}$$

sehingga menjadi

$$\mathbf{w}'\varphi(\mathrm{x}) = \mathrm{w}_1\mathrm{x}_1^2 + \sqrt{2}\mathrm{w}_2\mathrm{x}_1\mathrm{x}_2 + \mathrm{w}_3\mathrm{x}_2^2 \tag{2.29}$$

dengan dimensi pada *feature* space adalah kuadratik, padahal dimensi asalnya adalah *linear*. Metode kernel menghindari *learning* secara eksplisit memetakan data kedalam *feature* space dimensi tinggi, seperti pada contoh berikut.

$$\begin{aligned} f(x) &= \mathbf{w}'\mathbf{x} + b \\ &= \sum_{i=1}^{n}\alpha_i\mathbf{x}_i'\mathbf{x} + b \\ &= \sum_{i=1}^{n}\alpha_i\varphi(\mathbf{x}_i)'\varphi(\mathbf{x}) + b \text{ dalam } \textit{feature space } \mathbf{F} \\ &= \sum_{i=1}^{n}\alpha_i\mathrm{K}(\mathbf{x}_i,\mathbf{x}) + b \end{aligned}$$

Hubungan kernel dengan fungsi *mapping* adalah:

$$\begin{aligned} \varphi(\mathbf{x}_i)'\varphi(\mathbf{x}) &= \left(\mathrm{x}_{i1}^2, \sqrt{2}\mathrm{x}_{i1}\mathrm{x}_{i2}, \mathrm{x}_{i2}^2\right)\left(\mathrm{x}_1^2, \sqrt{2}\mathrm{x}_1\mathrm{x}_2, \mathrm{x}_2^2\right)' \\ &= \mathrm{x}_{i1}^2\mathrm{x}_1^2 + 2\mathrm{x}_{i1}\mathrm{x}_{i2}\mathrm{x}_1\mathrm{x}_2 + \mathrm{x}_{i2}^2\mathrm{x}_2^2 \\ &= (\mathbf{x}_i'\mathbf{x})^2 \\ &= K\ (\mathbf{x}_i, \mathbf{x}) \end{aligned}$$

Sedangkan untuk memperoleh fungsi klasifikasi non*linear* dalam data space, bentuk secara umumnya diperoleh dari penerapan *kernel trick* ke Persamaan (2.30):

$$L_D = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{2.30}$$

yaitu memaksimumkan $L_D : \max_{\alpha} L_D = \max \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$

dengan $\sum_{i=1}^{n} \alpha_i y_i = 0,\ 0 \le \alpha_i \le C;\ i = 1, 2, ..., n$

(Hardle & Simar, 2015)

Beberapa fungsi pembentuk matriks *kernel* yang umum digunakan pada SVM adalah:

1. Kernel *Linear*

   $K\left(\mathbf{x}_i, \mathbf{x}_j\right) = \mathbf{x}_i' \mathbf{x}_j$

2. Kernel Polinomial

   $K\left(\mathbf{x}_i, \mathbf{x}_j\right) = (\delta \mathbf{x}_i' \mathbf{x}_j + r)^p, \delta > 0$

3. Kernel *Radial Basis Function*(RBF)

   $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2\right), \gamma > 0$

4. Kernel Sigmoid

   $K\left(\mathbf{x}_i, \mathbf{x}_j\right) = \tanh(\delta \mathbf{x}_i' \mathbf{x}_j + r)$

Pemilihan fungsi kernel yang tepat merupakan hal yang sangat penting karena akan menentukan *feature* space dimana fungsi *classifier* akan dicari. Sepanjang fungsi kernelnya sesuai, SVM akan beroperasi secara benar meskipun tidak tahu pemetaan yang digunakan (Santosa, 2007). Menurut Hsu, Chang dan Lin (2004), fungsi kernel yang direkomendasikan untuk diuji pertama kali adalah fungsi kernel RBF karena dapat memetakan hubungan tidak *linear*, RBF lebih robust terhadap outlier karena fungsi kernel RBF berada antara selang $(-\infty, \infty)$ sedangkan fungsi kernel yang lain memiliki rentang antara (-1 sampai dengan 1). Selain itu menurut Scholkopf dan Simola (2002) fungsi kernel gaussian RBF mampu secara otomatis menentukan nilai, lokasi dari *center* dan nilai pembobot dan bisa mencakup rentang tak terhingga. Gaussian RBF juga efektif menghindari *overfitting* dengan memilih nilai yang tepat untuk parameter C dan γ dan RBF baik digunakan ketika tidak ada pengetahuan terdahulu.

2.3 *Entropy based Fuzzy Support Vector Machine* (EFSVM)

Evaluasi terhadap keanggotaan *fuzzy* adalah kunci dari EFSVM. Pada bagian ini akan dijelaskan dijelaskan terlebih dahulu mengenai keanggotaan *fuzzy*, kemudian cara menentukan keanggotaan *entropy based fuzzy* dan klasifikasi data menggunakan EFSVM pada data yang telah dinyatakan kedalam keanggotaan *entropy based fuzzy*.

2.3.1 *Fuzzy Membership*

Terdapat beberapa penerapan yang hanya ingin fokus pada akurasi untuk klasifikasi suatu kelas. Untuk tujuan tersebut dapat ditentukan keanggotaan *fuzzy* sebagai fungsi dari masing-masing kelas. Misalkan diberikan rangkaian *training*

$$(y_1, \mathbf{x}_1, s_1), ..., (y_n, \mathbf{x}_n, s_n) \tag{2.31}$$

Keanggotaan *fuzzy*, s_i, menjadi fungsi pada kelas y_i

$$s_i = \begin{cases} s_+, & \text{jika } y_i = 1 \\ s_-, & \text{jika } y_i = -1 \end{cases} \tag{2.32}$$

$$s_i = \begin{cases} 1, & \text{jika } y_i = 1 \\ 0.1, & \text{jika } y_i = -1 \end{cases} \tag{2.33}$$

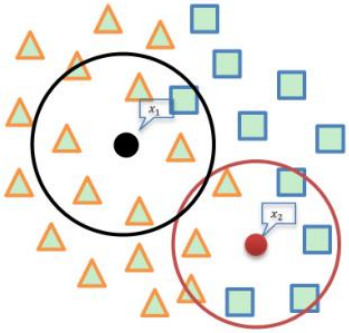
(Lin & Wang, 2002)

2.3.2 *Entropy based Fuzzy Membership*

Ketika berhadapan dengan *imbalanced* datasets, kelas yang positif akan lebih penting dari pada kelas yang negatif, sehingga harus lebih memperhatikan kelas positif dari pada kelas negatif. Pada tujuan evaluasi keanggotaan *entropy based fuzzy*, ditetapkan sampel positif maka keanggotaan *fuzzy* lebih besar misalnya 1,0 untuk menjamin kepentingan dari kelas positif. Untuk kelas negatif, kita membentuk keanggotaan *fuzzy* berdasarkan kepastian kelasnya (Fan, Wang, Li, Gao, & Zha, 2016).

Dalam teori informasi, *entropy* adalah rata-rata banyaknya informasi yang terkandung pada masing-masing pesan yang diterima (Shannon, 2001). *Entropy* menggambarkan kepastian tentang sumber informasi misalnya *entropy* yang lebih kecil mengindikasikan bahwa informasi tersebut lebih meyakinkan. Dengan

menggunakan *entropy*, kita dapat mengevaluasi kepentingan kelas dari sampel training. Maka kita menetapkan keanggotaan *fuzzy* pada sampel training berdasarkan kepastian kelasnya. Pada penerapannya, sampel dengan kepastian kelas yang lebih tinggi, misalnya *entropy* yang lebih rendah, ditetapkan untuk keanggotaan *fuzzy* yang lebih besar untuk menambah kontribusinya pada daerah keputusan begitu pula sebaliknya.



Gambar 2.5 Demonstrasi Evaluasi Probabilitas Kelas dengan 7 *Nearest Neigbors* (Diperoleh dari Fan, Wang, Li, Gao, & Zha, 2016)

Misalnya sampel training $\{\mathbf{x}_i, y_i\}_{i=1}^{N}, y_i \in \{+1,-1\}, y_i = +1$ menunjukkan bahwa sampel $\mathbf{x}_i$ merupakan anggota kelas positif, selain itu merupakan anggota kelas negatif. Peluang $\mathbf{x}_i$ masuk dalam kelas positif atau negatif masing-masing adalah p_{+i} dan p_{-i}. *Entropy* dari $\mathbf{x}_i$ didefinisikan sebagai berikut.

$$H_i = -p_{+i}\ln(p_{+i}) - p_{-i}\ln(p_{-i}) \tag{2.34}$$

Pada definisi *entropy* Persamaan (2.34) tujuan utama adalah untuk mengevaluasi peluang sampel masuk dalam masing-masing kelas. Sebagai informasi lokal dari sampel dapat direpresentasikan oleh neighbornya, evaluasi probabilitas berdasarkan pada *k nearest neighbors*. Misalnya untuk $\mathbf{x}_i$, pertama pilih k *nearest neighbors* nya $\{\mathbf{x}_{i1,\ldots,}\mathbf{x}_{ik}\}$. Kemudian hitunglah jumlah kelas positif dan negatif pada k sampel yang terpilih. Jumlah dari kelas positif dan negatif masing-masing dinyatakan dalam num_{+i} dan num_{-i}. Akhirnya peluang $\mathbf{x}_i$ masuk pada kelas positif dan negatif dihitung melalui persamaan:

$$p_{+i} = \frac{num_{+i}}{k} \tag{2.35}$$

$$p_{-i} = \frac{num_{-i}}{k} \tag{2.36}$$

Setelah mengevaluasi probabilitas kelas pada x_i kemudian dengan menggunakan Persamaan (2.34) maka dapat dihitung nilai *entropy*nya. Berikut ini diberikan contoh untuk menunjukkan bagaimana *entropy* merefleksikan kepastian dari sampel yang ditunjukkan pada Gambar 2.5 persegi dan segitiga menunjukkan masing-masing kelas positif dan kelas negatif. Digunakan 7 *nearest neigbors* untuk mengevaluasi peluang dari x_1 dan x_2 yang ditunjukkan oleh lingkaran. 7 *nearest neighbors* dari x_1 dan x_2 masuk dalam lingkaran tersebut. Selain itu, Tabel 2.1 menunjukkan data variabel yang yang sesuai dihitung untuk mengevaluasi *entropy* x_1 dan x_2. Didapatkan nilai $H_1 = 0.41$ dan $H_2 = 0.68$ yang mengindikasikan bahwa x_1 lebih pasti daripada x_2. Pada Gambar 2.5, x_2 lebih dekat dengan margin positif dan negatif daripada x_1. Seperti yang telah diketahui bahwa sampel yang lokasinya lebih dekat dengan margin maka lebih sensitif terhadap noise, yang menghasilkan kepastian yang lebih rendah pada kelas yang seharusnya sampel tersebut terdistribusi. Oleh karena itu dapat disimpulkan bahwa sampel yang lebih dekat denga margin memiliki kepastian kelas yang lebih rendah. Maka *entropy* dapat merefleksikan kepastian dari sampel yang mengindikasikan bahwa masuk akal untuk menggunakan *entropy* sebagai alat ukur kepastian sampel.

Tabel 2.1 Entopy dari Variabel yang Berhubungan

Sampel	num_+	num_-	p_+	p_-	**H**
x_1	6	1	6/7	1/7	0.41
x_2	4	3	4/7	3/7	0.68

Entropy untuk kelas negatif adalah $H = \{H_{-1}, H_{-2}, ..., H_{-n_-}\}$, n_- adalah jumlah sampel negatif. $H_{\min}$ dan $H_{\max}$ adalah nilai minimum dan maksimum dari *entropy H*. Keanggotaan *entropy based fuzzy* untuk sampel negatif dievaluasi

sebagai berikut. Pertama bagi negatif sampel kedalam *m* subset, misal $\{Sub_l\}_{l=1}^{m}$, dengan urutan peningkatan *entropy* berdasarkan nilai *entropy*nya seperti yang dijelaskan pada Tabel 2.1. $H_{Sub_1} < H_{Sub_2} < ... < H_{Sub_m}$, dimana H_{Sub_l} merupakan nilai *entropy* pada subset Sub_l.Kemudian keanggotaan *fuzzy* untuk sampel pada maisng-masing subset dinyatakan sebagai berikut

$$FM_l = 1.0 - \beta * (l-1), l = 1, 2, ..., \mathrm{m} \tag{2.37}$$

dimana FM_l adalah keanggotaan *fuzzy* untuk sampel yang didistribusikan pada subset Sub_l, parameter keanggotaan *fuzzy* $\beta \in \left(0, \frac{1}{m-1}\right]$ karena nilai FM_l adalah positif dan tidak lebih dari 1,0. Harus dideklarasikan bahwa sampel pada subset yang sama juga memiliki keanggotaan *fuzzy* yang sama sehingga mengindikasikan bahwa sampel yang terpilih dari subset yang sama memiliki kepentingan yang sama pada daerah keputusan. Disini akan ditunjukkan bagaimana $\beta \in \left(0, \frac{1}{m-1}\right)$ ditentukan. Keanggotaan *fuzzy* sampel FM_l pada masing-masing subset ditentukan oleh persaman (2.37). Karena nilai $FM_l \in [0,1]$ sehingga

$$0 \le \beta * (l-1) \le 1, \mathrm{l} = 1, 2, ..., \mathrm{m}$$

$$\begin{cases} \beta * (m-1) \le 1 \\ \beta \ge 0 \end{cases} \Rightarrow 0 \le \beta \le \frac{1}{m-1}$$

Ketika β dijadikan 0, keanggotaan *fuzzy* untuk semua sampel negatif sama dengan 1. misalnya semua sampel negatif memiliki kontribusi yang sama untuk training. Untuk menghindari kasus ini, β dibatasi lebih dari 0 sehingga $\beta \in (0, \frac{1}{m-1}]$. Dari persamaan diatas, kita dapat menemukan bahwa β ditentukan pada range FM_l:

$$1 - \beta * (m-1) \le FM_l \le 1$$

nilai β yang lebih besar menghasilkan range FM_l yang lebih lebar begitu pula sebaliknya. Untuk m=10, ambil $\beta = \frac{1}{9}$ dan $\beta = \frac{1}{18}$, *keanggotaan fuzzy* untuk

subset masing-masing adalah $\left\{1.0, \frac{8}{9}, \frac{7}{9}, ..., \frac{1}{9}, \frac{0}{9}\right\}$ dan. $\left\{1.0, \frac{17}{18}, \frac{16}{18}, ..., \frac{9}{18}\right\}$ sehingga β dapat mengontrol skala dari keanggotaan *fuzzy* pada sampel. Akhirnya keanggotaan *fuzzy* untuk training sampel x_i dinyatakan sebagai berikut:

$$s_i = \begin{cases} 1.0 & \text{if } y_i = +1 \\ FM_l & \text{if } y_i = +1 \& x_i \in Sub_l \end{cases} \tag{2.38}$$

For $l = 1: m$

$thrUp = H_{min} + \frac{l}{m}(H_{max} - H_{min})$

$thrlow = H_{min} + \frac{l-1}{m}(H_{max} - H_{min})$

For $i = 1: N\_$

$if\ thrLow \leq H_{\_i} < thrUp$

Sampel negatif x_i didistribusikan kedalam subset Sub_l

End

End

Gambar 2.6 Algoritma Pemisahan Sampel Negatif (Diperoleh dari Fan, Wang, Li, Gao, & Zha, 2016)

2.3.3 Klasifikasi dengan *Entropy-based Fuzzy Support Vector Machine* (EFSVM)

Dengan menggunakan evaluasi dari keanggotaan *entropy based fuzzy*, selanjutnya akan dijelaskan mengenai EFSVM. Diberikan training S, dimana $S = \{(x_i, y_i, s_i)\}_{i=1}^{N}$, x_i adalah sampel berukuran n, $y_i \in \{+1, -1\}$ yang menyatakan kelas (+1 untuk kelas positif dan -1 untuk kelas negatif), dan s_i adalah keanggotaan *entropy based fuzzy* yang ditentukan oleh Persamaan (2.38). EFSVM menemukan daerah keputusan optimal yang membagi kelas positif dan negatif dengan margin sebesar mungkin. Untuk menemukan keputusan yang optimal maka perlu untuk menyederhanakan masalah optimasi kuadrat berikut.

$$\min \quad \tfrac{1}{2}\mathbf{w}'\mathbf{w} + C\sum_{i=1}^{N} s_i \xi_i \tag{2.39}$$

$$y_i(\mathbf{w}'\varphi(\mathbf{x}_i) + b) \geq 1 - \xi_i,\ \xi_i \geq 0,\ i = 1, ..., N \tag{2.40}$$

dimana $\mathbf{w}$ adalah vektor pembobot pada daerah keputusan, *b* menyatakan bias, $\varphi(\mathbf{x}_i)$ merupakan fungsi nonlinear yang memetakan $\mathbf{x}_i$ kedalam ruang *feature high dimensional* di mana daerah keputusan yang lebih baik dapat ditemukan, C adalah parameter regularisasi yang dipilih terlebih dahulu untuk mengontrol *trade-off* antara margin klasifikasi dan biaya kesalahan klasifikasi. Variabel non-negatif ξ_i menyatakan variabel *slack* dari $\mathbf{x}_i$ pada SVM, sedangkan $s_i\xi_i$ adalah ukuran error dengan bobot yang berbeda sesuai dengan s_i.

Untuk mengatasi optimasi kuadratik, Persamaan (2.39) dinyatakan kedalam lagrangian pada Persamaan (2.41).

$$\begin{aligned} L_P(\mathbf{w},b,\xi,\alpha,\mu) = \frac{1}{2}\mathbf{w}'\mathbf{w} + C\sum_{i=1}^{N} s_i\xi_i \\ -\sum_{i=1}^{N}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i) - \sum_{i=1}^{N}\mu_i\xi_i \end{aligned} \tag{2.41}$$

dan untuk mendapatkan saddle point dari $L(\mathbf{w},b,\xi,\alpha,\mu)$ parameter harus memenuhi kondisi KKT.

$$\frac{\partial L(\mathbf{w},b,\xi,\alpha,\mu)}{\partial w} = 0 \tag{2.42}$$

$$\frac{\partial L(\mathbf{w},b,\xi,\alpha,\mu)}{\partial b} = 0 \tag{2.43}$$

$$\frac{\partial L(\mathbf{w},b,\xi,\alpha,\mu)}{\partial \xi_i} = 0 \tag{2.44}$$

$$\begin{aligned} &\alpha_i(y_i(\mathbf{w}^{\mathbf{T}}\mathbf{x}_{\mathbf{i}}+b)-1+\xi_i) = 0, \ i=1,...,N, \\ &\mu_i\xi_i = 0, \qquad\qquad i=1,...,N, \\ &\alpha_i \geq 0, \mu_i \geq 0, \xi_i \geq 0, \qquad i=1,...,N, \end{aligned} \tag{2.45}$$

Dengan menyelesaikan persamaan (2.42) sampai (2.44) maka didapatkan hasil untuk masing-masing kondisi KKT sebagai berikut.

$$\mathbf{w} = \sum_{i=1}^{N}\alpha_i y_i \varphi(x_i) \tag{2.46}$$

$$\sum_{i=1}^{N}\alpha_i y_i = 0 \tag{2.47}$$

$$s_i C - \alpha_i - \mu_i = 0 \tag{2.48}$$

Terapkan kondisi KKT tersebut pada lagrangian Persamaan (2.41) sehingga didapatkan dual problem pada Persamaan (2.49).

$$\max \ \mathrm{L}_D(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \varphi(\mathbf{x}_i) \varphi(\mathbf{x}_j) \tag{2.49}$$

$$\begin{aligned} &\sum_{i=1}^{N} \alpha_i y_i = 0 \\ &0 \le \alpha_i \le s_i C \\ &i = 1, \ldots, N \end{aligned} \tag{2.50}$$

Nilai optimasi untuk α_i didapatkan dengan menyelesaikan Persamaan (2.49) dan vektor pembobot **w** dapat dihitung sebagai berikut.

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \varphi(\mathbf{x}_i) \tag{2.51}$$

selanjutnya, fungsi keputusan untuk EFSVM dinyatakan sebagai berikut

$$f(x) = sign(\sum_{i=1}^{N} \alpha_i y_i \varphi(\mathbf{x}_i)' \varphi(\mathbf{x}) + b) \tag{2.52}$$

dimana bias *b* didapatkan dengan Karush-Kuhn-Tucker (KKT). Perbedaan penting antara SVM dan FSVM adalah bahwa titik-titik dengan nilai α_i yang sama dapat mengindikasikan jenis *suport vector* yang berbeda pada FSVM karena adanya faktor s_i.

2.4 Evaluasi Performansi Metode Klasifikasi

Performa klasifikasi menunjukkan kemampuan metode klasifikasi untuk memprediksikan kelas suatu data. Hasil dari klasifikasi dapat disusun dalam sebuah *confusion matrix* berikut.

Tabel 2.2 *Confusion Matrix*

Kelas Aktual	Kelas Prediksi	
	Positif	Negatif
Positif	TP	FN
Negatif	FP	TN

Keterangan:

TP : Banyaknya prediksi benar pada kelas positif

FP : Banyaknya prediksi salah pada kelas positif

TN : Banyaknya prediksi benar pada kelas negatif

FN : Banyaknya prediksi salah pada kelas negatif

Dari *confusion matrix* tersebut dapat dihitung nilai akurasi, sensitivity, dan specificity. Selain itu, performa klasifikasi juga diukur melalui ukuran performa klasifikasi yang relevan pada data *imbalance,* yaitu G-*mean*, dan *Area Under* ROC *Curve* (AUC).

1) Akurasi

Akurasi menilai keseluruhan efektivitas algoritma dengan memperkirakan probabilitas nilai benar dari label kelas. Nilai Akurasi dinyatakan sebagai berikut.

$$\text{Akurasi}=\frac{\text{TN+TP}}{\text{TN+TP+FN+FP}} \tag{2.53}$$

2) *Sensitivity*

Sensitivity adalah ukuran kelengkapan atau keakuratan dari sampel positif. Nilai sensitivity menyatakan berapa banyak sampel kelas positif yang diberi label dengan benar. Nilai *sensitivity* dinyatakan sebagai berikut.

$$\textit{Sensitivity}=\frac{\text{TP}}{\text{TP+FN}} \tag{2.54}$$

3) *Specificity*

Berbeda dengan *sensitivity*, *specificity* menilai keefektifan algoritma pada kelas negatif. Nilai *specificity* menyatakan berapa banyak sampel kelas negatif yang diberi label dengan benar. Nilai *specificity* dinyatakan sebagai berikut.

$$\textit{Specificity}=\frac{\text{TN}}{\text{TN+FP}} \tag{2.55}$$

4) *G-means*

G-means adalah sebagai perkalian dari akurasi prediksi untuk kedua kelas, yang meliputi, akurasi pada kelas positif (*sensitivity*) dan akurasi pada kelas negatif (*specificity*). Nilai ini menunjukkan keseimbangan antara kinerja klasifikasi pada kelas mayoritas dan minoritas. kinerja yang buruk dalam prediksi sampel positif akan menghasilkan nilai rata-rata G yang rendah begitu pula untuk kelas negatif. Nilai *g-means* dinyatakan sebagai berikut.

$$G\text{-}mean=\sqrt{Sensitivity\times Spesificity} \tag{2.56}$$

5) Area Under ROC Curve (AUC)

Secara grafik sering digunakan ROC (*Receiver Operating Characteristic*) sebagai kurva yang memberikan TP (*True Positive*) rate sebagai fungsi dari FP (*False Positive*) rate pada kelompok yang sama. Pendekatan ROC mewakili nilai *sensitivity* sebagai fungsi dari (1-*specificity*) untuk semua kemungkinan nilai ambang dan menggabungkan titik-titik dengan sebuah kurva (Bekkar, Djema, & Alitouche, 2013). Area dibawah kurva ROC (Area Under Curve, AUC) adalah indikator ringkasan dari performansi kurva ROC yang dapat merangkum performansi pada suatu classifier kedalam metrik tunggal. Tidak seperti kesulitan yang ditemui pada perbandingan dari kurva ROC yang berbeda khususnya pada kasus kurva ROC dengan interseksi, AUC dapat mengurutkan model berdasarkan performansi keseluruhan, sebagai hasilnya, AUC lebih dipertimbangkan pada penilaian model (Batista, Ronaldo, & Monard, 2004). AUC diestimasi melalui beberapa teknik, yang paling banyak digunakan adalah metode *trapezoidal* yang merupakan metode geometrik berdasarkan interpolasi *linear* antara masing-masing point pada kurva ROC. Bekkar dkk (2013) mengusulkan untuk membuat pendekatan AUC dalam kasus *binary learning* dengan *Balanced Accuracy* yang ditunjukkan pada Persamaan (2.57)

$$Balanced\ Accuracy = \frac{1}{2}\left(sensitivity + specificity\right) = \frac{TP}{P} + \frac{TN}{N} \qquad (2.57)$$

Dalam prakteknya, nilai AUC bervariasi antara 0,5 dan 1.

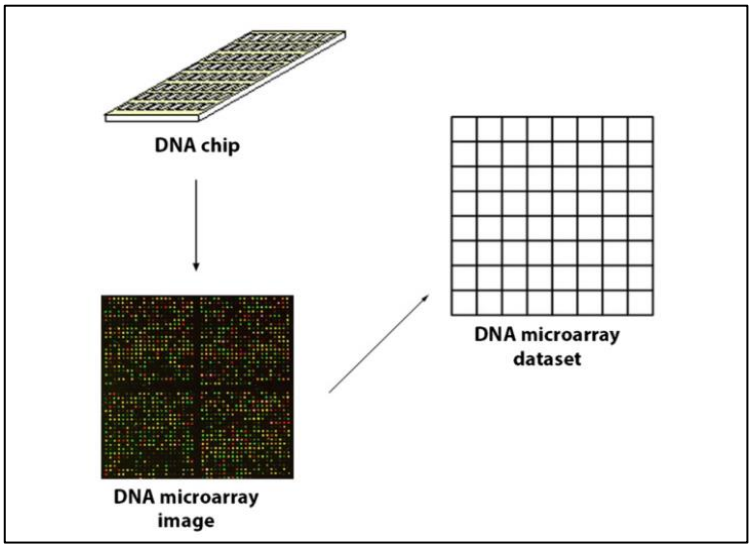
2.5 *Stratified* K-*Fold Cross Validation*

Cross validation adalah metode statistik untuk mengevaluasi dan membandingkan algoritma pembelajaran dengan membagi data menjadi dua bagian yaitu data training yang digunakan untuk training dan data testing yang digunakan untuk memvalidasi model (Refaeilzadeh, Tang, dan Liu, 2008). K-fold *cross validation* akan membagi data ke dalam k subset yang saling bebas yaitu S_1, S_2, ..., S_k dengan jumlah data tiap subset hampir sama, selanjutnya jika satu subset menjadi data testing maka *k-1* subset yang akan menjadi data training (Han, Kamber, & Jian, 2006). Data biasanya distratifikasi sebelum dipecah kedalam k-*fold*. Stratifikasi adalah proses penyusunan ulang data untuk memastikan setiap

fold merupakan representasi yang baik dari keseluruhan data. Misalnya dalam masalah klasifikasi biner dimana masing-masing kelas terdiri dari 50% data, cara yang terbaik adalah dengan mengatur data sedemikian rupa sehingga dalam setiap *fold*, setiap kelasnya terdapat sekitar setengah sampel (Refaeilzadeh, Tang, dan Liu, 2008).

2.6 Data *Microarray*

Selama dua dekade terakhir, munculnya dataset *microarray* DNA telah mendorong inovasi baru dalam penelitian baik dibidang bioinformatika dan *machinelearning*. Semua sel memiliki inti, dan di dalam inti ini ada DNA, yang mengkode "program" untuk organism masa depan. DNA mempunyai *coding* dan *non-coding* segmen.Segmen *coding*, juga dikenal sebagai gen, menentukan struktur protein, yang melakukan pekerjaan penting dalam setiap organisme. Gen membuat protein dalam dua langkah: DNA ditranskripsi menjadi mRNA dan kemudian mRNA diterjemahkan menjadi protein. Kemajuan dalam teknologi genetika molekuler, seperti DNA *microarray*, memungkinkan kita untuk memperoleh pandangan global terhadap sel, yang mana dimungkinkan untuk mengukur ekspresi simultan dari puluhan ribu gen (Shapiro & Tamayo, 2003). Gambar 2.7 menampilkan proses umum memperoleh data ekspresi gen dari *microarray* DNA.



Gambar 2.7 Proses Umum Memperoleh Data Ekspresi Gen DNA Microarray (Diperoleh dari Canedo, Marono, Betanzos, Benitez, & Herrera, 2014)

Jenis data ini digunakan untuk mengumpulkan informasi dari jaringan dan sel sampel mengenai perbedaan ekspresi gen yang dapat berguna untuk diagnosis penyakit atau untuk membedakan jenis tertentu dari tumor. Klasifikasi data *microarray* menimbulkan tantangan serius bagi teknik komputasi, karena dimensi yang besar (hingga beberapa puluhan ribu gen) dengan ukuran sampel yang kecil. Masalah umum dalam data *microarray* adalah yang disebut masalah ketidakseimbangan kelas. Hal ini terjadi ketika sebuah dataset didominasi oleh kelas utama atau kelas yang telah secara signifikan lebih banyak contoh dari kelas langka/ minoritas lainnya dalam data. Biasanya, orang memiliki lebih minat belajar kelas langka. Misalnya, dalam domain, kelas kanker cenderung jarang daripada kelas non-kanker karena biasanya ada pasien lebih sehat. Namun, penting bagi praktisi untuk memprediksi dan mencegah munculnya kanker. Dalam kasus ini, standar belajar *classifier* algoritma memiliki bias terhadap kelas dengan jumlah yang lebih besar dari kasus, karena aturan-aturan yang benar memprediksi contoh-contoh yang positif tertimbang mendukung akurasi metrik, sedangkan aturan khusus yang memprediksi contoh dari kelas minoritas yang biasanya diabaikan (diperlakukan sebagai *noise*), karena aturan yang lebih umum lebih disukai. Oleh karena itu, contoh kelas minoritas lebih sering kesalahan klasifikasi daripada yang dari kelas-kelas lain (Canedo, Marono, Betanzos, Benitez, & Herrera, 2014)

BAB 3
METODOLOGI PENELITIAN

3.1 Sumber Data

Studi kasus yang digunakan dalam penelitian ini adalah *binary classification* untuk data *imbalanced* DNA *microarray*. Data yang digunakan didapatkan dari *http://datam.i2r.a-star.edu.sg/datasets/krbd*. Deskripsi dari data yang digunakan ditunjukkan pada Tabel 3.1.

Tabel 3.1 Deskripsi Data DNA *Microarray*

Data	Sample	*Feature*	Distribusi kelas	IR*
Breast Cancer	168	2905	111 *good* & 57 *poor*	1,9
Colon Cancer	62	2000	40 tumor (t) & 22 normal (n)	1,8

*IR = jumlah data kelas negatif/jumlah data kelas positif

a. *Breast Cancer Datasets*

Data *breast cancer* didapatkan dari hasil pengamatan selama 5 tahun yang dilakukan untuk mengamati terjadinya *event* yaitu *metastasis* pada sel kanker. Data tersebut terdiri dari 2905 *feature* dan 168 sampel yang terbagi dalam dua kelas. Kelas pertama adalah kelas "*good*" yang berarti selama pengamatan tidak terjadi *event*, dan kelas kedua adalah kelas "*poor*" yang berarti selama kurun waktu 5 tahun telah terjadi *event*. Imbalanced Ratio (IR) adalah perbandingan antara kelas "*good*" dan "*poor*" yang didapatkan dengan membagi jumlah sampel pada kelas "*good*" dengan kelas "*poor*".

b. *Colon Cancer Datasets*

Colon cancer atau kanker usus besar adalah tumbuhnya sel-sel ganas di permukaan dalam usus besar (kolon) atau rektum. Data ini terdiri dari 62 sampel yang terbagi ke dalam dua kelas, yaitu 22 data normal biopsies (n) dan kelas kedua terdiri dari 40 data tumor biopsies yang diberi label (t) dimana masing-masing unit sampel terdiri dari 2000 ekspresi gen. IR pada data *colon cancer* didapatkan dengan membagi jumlah sampel pada kelas "t" dengan kelas "n".

3.2 Struktur Data

Berikut ini diberikan struktur data untuk masing-masing data. Struktur data *breast cancer* ditunjukkan pada Tabel 3.2

Tabel 3.2 Struktur Data *Breast Cancer*

Sampel	g2E09	g7F07	g1A01	...	g6D07	Kelas
1	0.2125	-0.3803	-0.3196	...	-0.6808	*Good*
2	-0.2631	-0.2818	0.3060	...	0.298258	*Good*
3	0.5455	-0.0113	-0.2846	...	-0.97952	*Good*
...	...	...	...	...	...	...
111	-0.53323	0.21477	-0.3666	...	0.9363	*Good*
112	0.0482	-0.277	0.5683	...	0.5828	*Poor*
...	...	...	...	...	...	...
168	1.07674	1.019438	1.390224	...	0.8139	*Poor*

Struktur data *colon cancer* ditunjukkan pada Tabel 3.3

Tabel 3.3 Struktur Data *Colon Cancer*

Sampel	X1	X2	X3	...	X2000	Kelas
1	0.508777	0.229011	0.09278	...	-0.50242	Tumor
2	0.694628	0.800666	0.433821	...	-0.92256	Tumor
3	-1.0314	0.915258	0.701422	...	-0.97925	Tumor
...	...	...	...	...	...	...
40	-1.04	0.319257	0.58432	...	-0.19427	Tumor
41	-0.26561	0.107378	-0.0083		-1.24726	Normal
...	...	...	...	...	...	...
62	0.147503	-0.59986	-0.75162	...	-0.1174	Normal

3.3 Tahapan Penelitian

Langkah-langkah yang dilakukan berkaitan dengan tujuan penelitian adalah sebagai berikut:

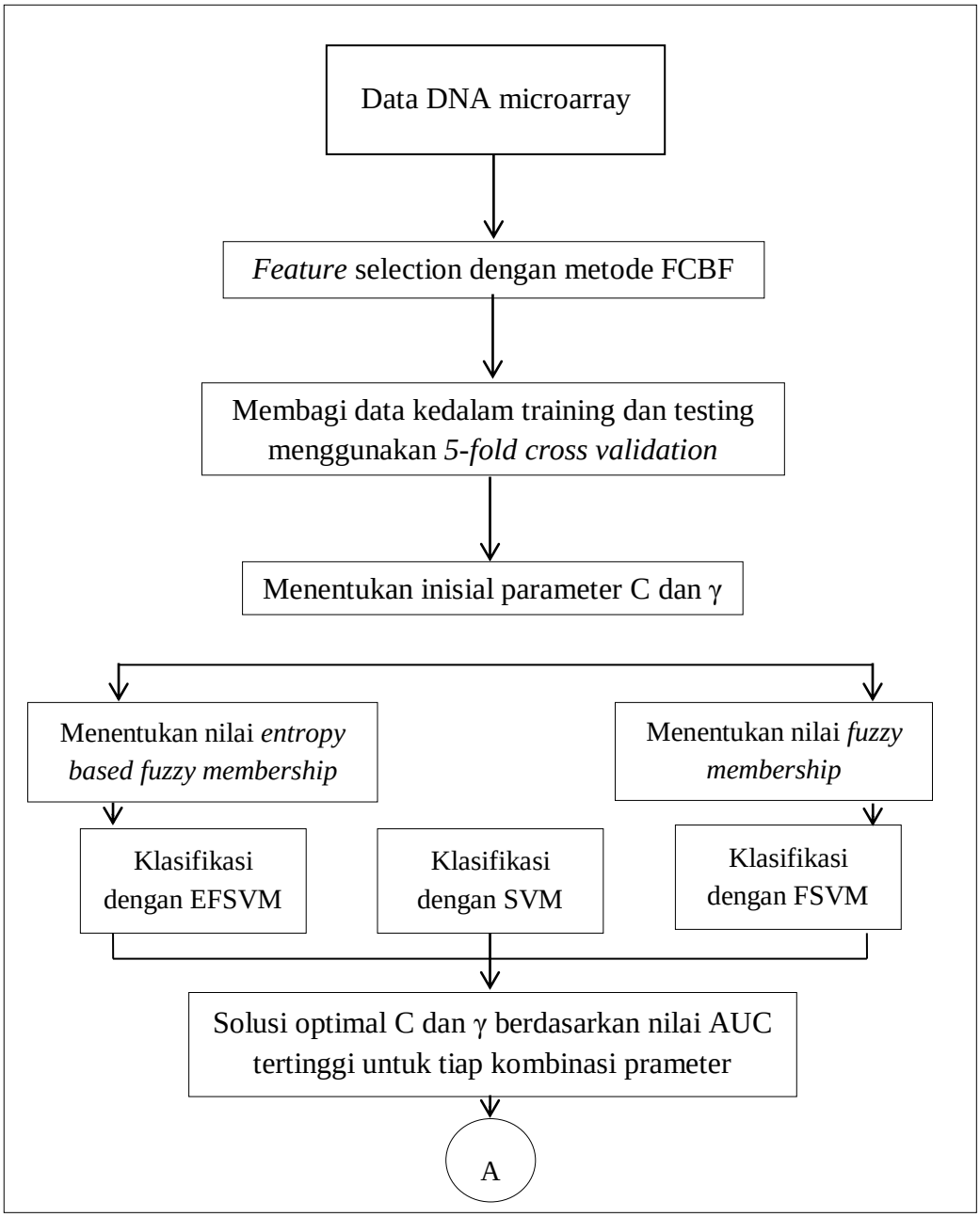
1. Kajian teoritis metode *Entropy Based Fuzzy Support Vector Machine* (EFSVM)
   a. Menyelesaikan permasalahan optimasi kuadrat dengan membangun fungsi lagrange dari fungsi tujuan pada persamaan (2.39) dan fungsi kendala pada Persamaan (2.40)
   b. Menurunkan fungsi lagrange (2.41) terhadap $\mathbf{w}, b,$ dan ξ pada kondisi Karush Kuhn Tucker (KKT) yang hasilnya ditunjukkan pada Persamaan (2.46), (2.47), dan (2.48)

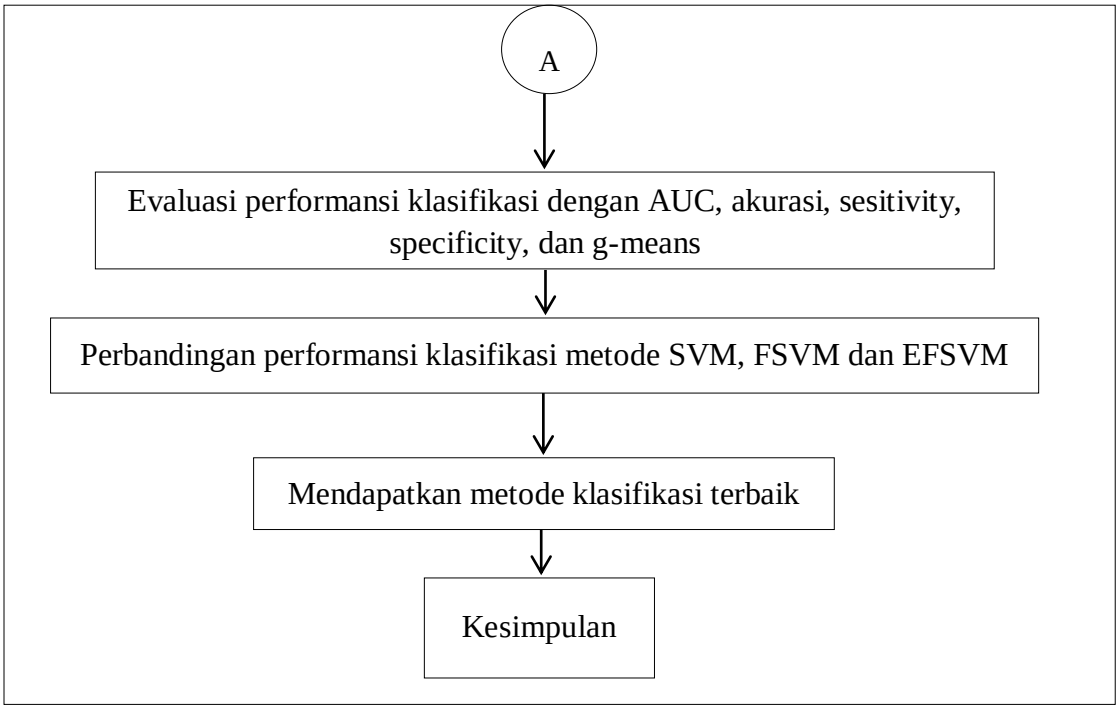
c. Mensubstitusikan hasil kondisi KKT pada fungsi lagrange *primal problem* (2.41) sehingga didapatkan lagrangian untuk dual problem pada Persamaan (2.49)
d. Menyelesaikan lagrangian dual problem dengan *quadratic programming* (QP) solver sehingga didapatkan $\boldsymbol{\alpha}$.
e. Mendapatkan fungsi keputusan pada persamaan (2.52)

2. Melakukan proses *feature selection* untuk setiap data *microarray* menggunakan metode FCBF
3. Analisis data *imbalanced* DNA *microarray* menggunakan SVM, FSVM dan EFSVM dengan langkah-langkah sebagai berikut.
   a. Membagi data ke dalam data *training* dan *testing* menggunakan *stratified* 5-*fold cross validation*
   b. Menentukan *fuzzy membership* pada training data untuk FSVM menggunakan Persamaan (2.33)
   c. Menentukan *entropy based fuzzy membership* pada training data untuk EFSVM yang meliputi,
      1. Menentukan *k-Nearest Neighbors* (k-NN) untuk masing-masing sampel
      2. Menghitung jumlah sampel positif (num_{+i}) dan negatif (num_{-i}) pada k-NN
      3. Menghitung peluang dari sampel positif dan negatif menggunakan Persamaan (2.35) dan (2.36)
      4. Menghitung *entropy* dari dari masing-masing training data menggunakan Persamaan (2.34)
      5. Membagi sampel negatif kedalam *m* subset berdasarkan Algoritma pada Gambar 2.6
      6. Menetapkan *entropy based fuzzy membership* untuk masing-masing sampel berdasarkan Persamaan (2.38)
4. Mendapatkan nilai optimasi parameter σ dan C terbaik berdasarkan nilai AUC (%) yang dihitung menggunakan Persamaan (2.57). *Range* parameter gamma (γ) dan parameter cost (C) yang akan dioptimasi yaitu C = 2^1, 2^2 ,

2^{3}, 2^{4}, ..., 2^{11} dan $\gamma = 2^{-18}$, 2^{-17}, 2^{-16}, 2^{-15},..., 2^{-4}. Cara penentuan range ini berdasarkan aturan grid-search pada LIBSVM, Chang dkk. (2016) merekomendasikan identifikasi parameter C dan γ menggunakan *exponentially growing sequences* misal 2^{-5}, 2^{-3}, ..., 2^{15}.

5. Melakukan perbandingan performansi hasil klasifikasi dengan metode SVM, FSVM, dan EFSVM berdasarkan nilai rata-rata AUC, akurasi, *sensitivity*, *specificity*, dan *g-means*
6. Menarik kesimpulan.





Gambar 3.1 Tahapan Penelitian

(Halaman ini sengaja dikosongkan)

BAB 4
ANALISIS DAN PEMBAHASAN

Pada bab ini dibahas mengenai kajian teori *fuzzy* SVM yakni memberikan penjabaran bagaimana perumusan ulang SVM dengan menambahkan keanggotaan *fuzzy* hingga didapatkan fungsi pemisah. Selanjutnya ditunjukkan pula hasil *feature selection* pada data *microarray* menggunakan FCBF dilanjutkan dengan hasil klasifikasi data *microarray* menggunakan metode SVM, FSVM, dan EFSVM yang dinyatakan dalam performansi klasifikasi yang meliputi akurasi, *sensitivity*, *specificity*, *G-means* dan AUC.

4.1 Kajian Teoritis EFSVM

Pada bagian ini dijelaskan mengenai algoritma dari keanggotaan *entropy based fuzzy* kemudian teori SVM dengan penambahan *fuzzy* didalamnya.

4.1.1 Algoritma Keanggotaan *Entropy based Fuzzy*

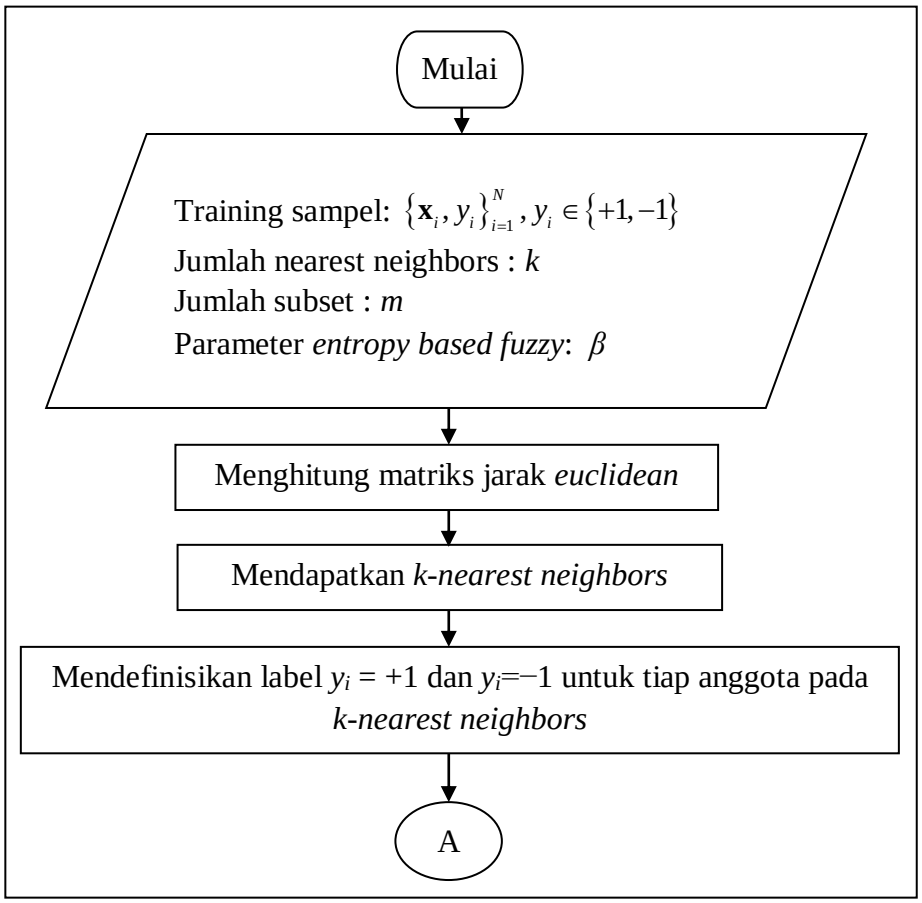
Pada bagian ini dijelaskan cara mendapatkan keanggotaan *entropy based fuzzy*.Berikut ini merupakan algoritma dari keanggotaan *entropy based fuzzy*.

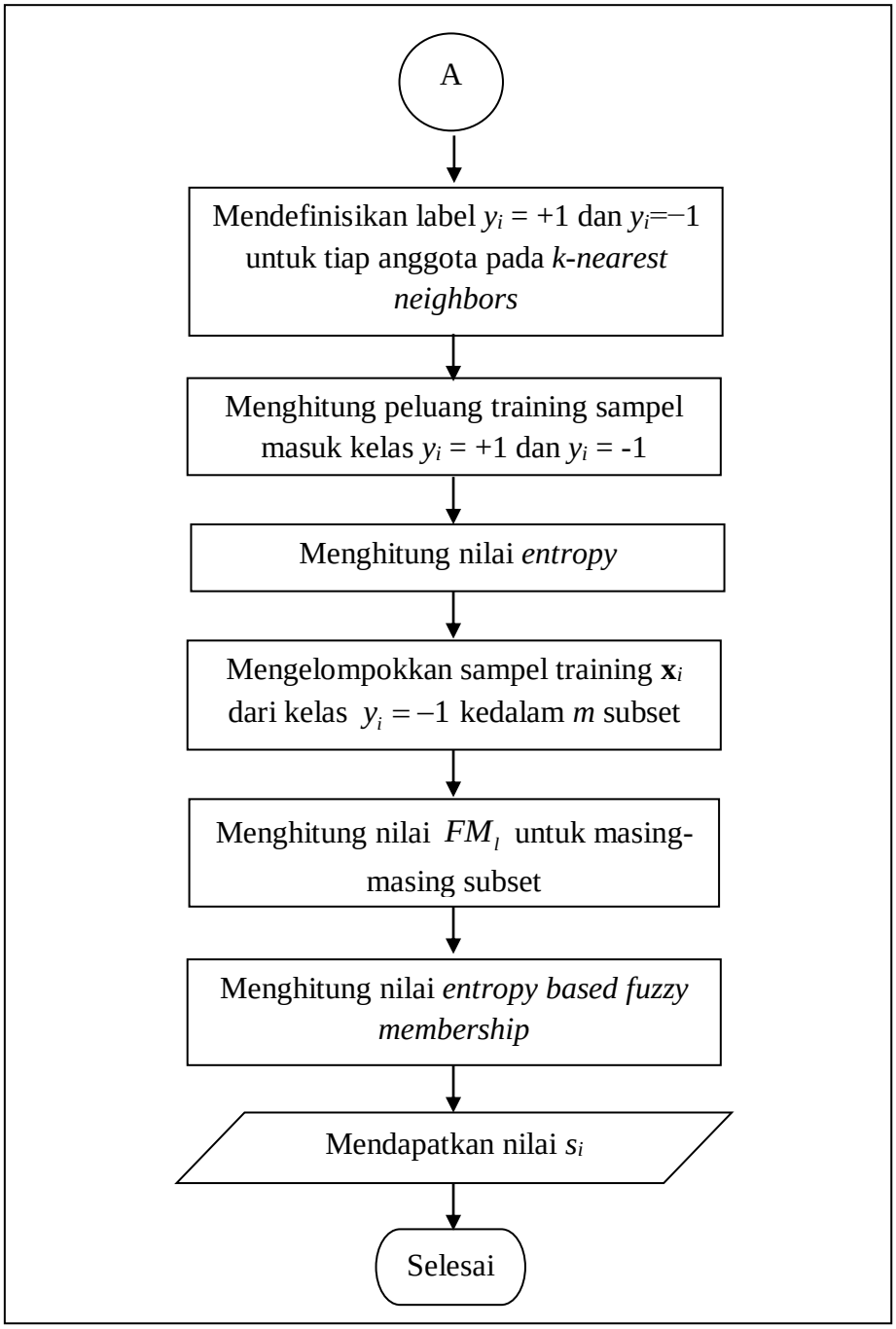
1. Menghitung matriks jarak menggunakan metode *euclidean* berdasarkan nilai *feature* pada masing-masing sampel training x_i.
2. Mendapatkan *k-nearest neighbors* berdasarkan nilai jarak yang paling dekat. Pada penelitian digunakan k=7 merujuk pada penelitian sebelumnya oleh Fan, Wang, Li, Gao, & Zha (2016).
3. Mendefinisikan label $y_i = +1$ atau $y_i = -1$untuk tiap anggota pada *k-nearest neighbor* dan menghitung peluang masuk pada kelas $y_i = +1$ menggunakan persamaan (2.35) dan masuk kelas $y_i = -1$ menggunakan persamaan (2.36).
4. Menghitung nilai *entropy* menggunakan persamaan (2.34)
5. Ulangi langkah 2 sampai 5 untuk seluruh sample training $\mathbf{x}_i$.
6. Pada sampel training $\mathbf{x}_i$ dari kelas $y_i = -1$dilakukan pengelompokan kedalam *m* subset. Untuk mengelompokkan digunakan algoritma pada Gambar 2.6. Berdasarkan Gambar 2.6, *m* merupakan jumlah subset, *thrUp* adalah batas atas untuk masing-masing subset yang nilainya adalah $thrUp = H_{min} + \frac{1}{m}(H_{max} - H_{min})$. H_{min} dan H_{max} masing-masing adalah nilai *entropy* paling

kecil dan besar yang telah dihitung pada langkah 5. *thrLow* adalah nilai batas bawah untuk masing-masing subset yang nilainya adalah $thrlow = H_{min} + \frac{l-1}{m}(H_{max} - H_{min})$. Setelah didapatkan nilai *thrUp* dan *thrLow* pada masing-masing subset maka sample training dari kelas $y_i = -1$ dikelompokkan berdasarkan nilai *entropy*.

7. Menghitung nilai *fuzzy membership* FM_l untuk masing-masing subset menggunakan persamaan (2.37).
8. Menentukan nilai *entropy based fuzzy membership* untuk semua training sampel x_i menggunakan persamaan (2.38). Sampel dari kelas $y_i = +1$ diberi nilai s_i sama dengan 1 sedangkan untuk sampel dari kelas $y_i = -1$ nilai s_i akan sesuai dengan yang telah dihitung pada langkah 7.

Lebih jelasnya diberikan diagram alir *entropy based fuzzy membership* pada Gambar 4.1.





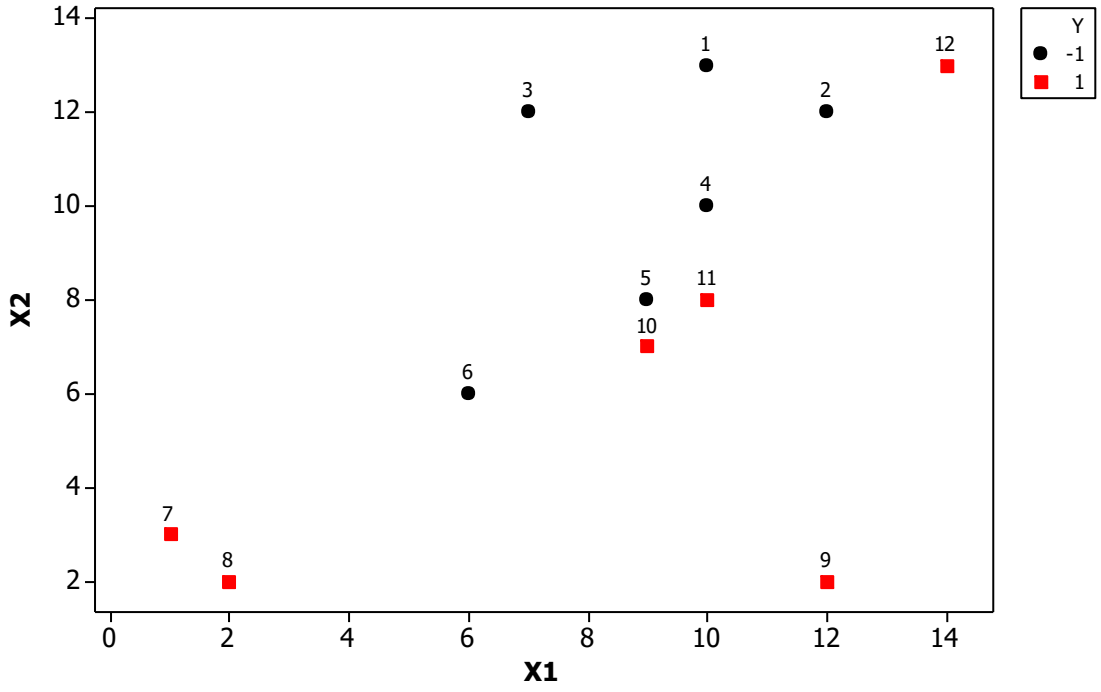
Gambar 4.1 Diagram Alir *Entropy Based Fuzzy Membership*

Untuk lebih jelasnya, berikut diberikan ilustrasi bagaimana mendapatkan nilai *entropy based fuzzy membership*. Misal pada Tabel 4.1 diberikan 12 data yang terdiri dari 6 data kelas positif dan 6 data kelas negatif dengan 2 variabel yaitu x_1 dan x_2.

Tabel 4.1 Data Ilustrasi

Data	x_1	x_2	y
1	10	13	-1
2	12	12	-1
3	7	12	-1
4	10	10	-1
5	9	8	-1
6	6	6	-1
7	1	3	1
8	2	2	1
9	12	2	1
10	9	7	1
11	10	8	1
12	14	13	1

Plot dari 12 data tersebut ditunjukkan pada Gambar 4.1. Warna merah menunjukkan data dari kelas positif sedangkan warna hitam menunjukkan data dari kelas positif.



Gambar 4.2 *Scatterplot* Data Ilustrasi

Berdasarkan Gambar 4.2 dapat dilihat bagaimana kedekatan satu unit data dengan data yang lain. Langkah selanjutnya yaitu mendapatkan matriks jarak dari 12 data yang ditunjukkan pada Tabel 4.2.

Tabel 4.2 Matriks Jarak Data Ilustrasi

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	2	3.16	3	5.09	8.06	13.45	13.6	11.8	6.08	5	4
2	2	0	5.09	3.6	5.83	9.21	14.86	14.86	11	6.7	5.38	2
3	3.16	5.09	0	3.6	4.4	6.08	10.81	11.18	11.18	5.38	5	7.07
4	3	3.6	3.6	0	2.23	5.65	11.4	11.3	8.24	3.16	2	5
5	5.09	5.83	4.4	2.23	0	3.6	9.4	9.2	6.7	1	1	7.07
6	8.06	9.21	6.08	5.65	3.6	0	5.83	5.65	7.2	3.16	4.47	10.63
7	13.45	14.86	10.81	11.4	9.4	5.83	0	1.41	11	8.9	10.2	16.4
8	13.6	14.86	11.18	11.3	9.2	5.65	1.41	0	10	8.6	10	16
9	11.8	11	11.18	8.24	6.7	7.2	11	10	0	5.8	6.3	11
10	6.08	6.7	5.38	3.16	1	3.16	8.9	8.6	5.8	0	1.4	7.8
11	5	5.38	5	2	1	4.47	10.2	10	6.3	1.4	0	6.4
12	4	2	7.07	5	7.07	10.63	16.4	16	11	7.8	6.4	0

Melalui matriks jarak dapat diketahui kedekatan suatu data dengan data yang lain. Setelah didapatkan matriks jarak selanjutnya yaitu mendapatkan nilai *entropy*. Langkah-langkah mendapatkan *entropy* berikut hasilnya ditunjukkan pada Tabel 4.3. Berdasarkan Tabel 4.3 kolom k-NN merupakan 7 data dengan jarak terdekat yang diukur berdasarkan matriks jarak pada Tabel 4.2. Angka yang dicetak tebal pada kolom tersebut menunjukkan unit sampel tersebut berasal dari kelas positif sedangkan yang tidak tercetak tebal merupakan unit sampel dari kelas negatif. Berdasarkan jumlah unit sampel dari kelas negatif dan positif kemudian dihitung P_{+i} dan P_{-i} yanghasilnya ditampilkan pada kolom ke-3 dan ke-4. Dari nilai P_{+i} dan P_{-i} kemudian dapat dihitung nilai *entropy* menggunakan persamaan yang hasilnya ditunjukkan pada kolom terakhir pada Tabel 4.3.

Tabel 4.3 Nilai *Entropy* Data Ilustrasi

Data	*k*-NN							P_{+i}	P_{-i}	$\ln(P_{+i})$	$\ln(P_{-i})$	H
1	2	4	3	**12**	**11**	5	**10**	0.428	0.571	-0.847	-0.559	0.682
2	1	**12**	4	**11**	5	**10**	6	0.428	0.571	-0.847	-0.559	0.682
3	1	4	5	**11**	**10**	6	**12**	0.428	0.571	-0.847	-0.559	0.682
4	1	2	3	5	**11**	**12**	6	0.285	0.714	-1.252	-0.336	0.598
5	**10**	**11**	4	6	3	**9**	**12**	0.428	0.571	-0.847	-0.559	0.682
6	5	**10**	**11**	6	**8**	4	**7**	0.428	0.571	-0.847	-0.559	0.682
7	**8**	6	**10**	5	**11**	**9**	4	0.571	0.428	-0.559	-0.847	0.682
8	**7**	6	5	**9**	**11**	4	3	0.428	0.571	-0.847	-0.559	0.682
9	**10**	**11**	5	6	4	**8**	2	0.428	0.571	-0.847	-0.559	0.682
10	5	**11**	4	6	3	**9**	1	0.285	0.714	-1.252	-0.336	0.598
11	5	**10**	4	6	3	1	2	0.142	0.857	-1.945	-0.154	0.410
12	2	1	4	**11**	5	**10**	6	0.285	0.714	-1.252	-0.336	0.598

Setelah didapatkan nilai *entropy* selanjutnya mengelompokkan unit sampel dari kelas negatif kedalam subset-subset. Untuk mengelompokkan sampel negatif ke dalam subset-subset, terlebih dahulu didapatkan nilai maksimum dan minimum *entropy* masing-masing yaitu, 0,410116 dan 0,6829. Selanjutnya dihitung batas bawah dan batas atas masing-masing subset sebagai berikut.

$$thrlow_1 = H_{\min} + \frac{1-1}{m}\left(H_{\max} - H_{\min}\right) = 0,410116 + \frac{0}{5}\left(0,6829 - 0,410116\right) = 0,410116$$

$$thrlow_2 = H_{\min} + \frac{2-1}{m}\left(H_{\max} - H_{\min}\right) = 0,410116 + \frac{1}{5}\left(0,6829 - 0,410116\right) = 0,464675$$

$$\vdots$$

$$thrlow_5 = H_{\min} + \frac{5-1}{m}\left(H_{\max} - H_{\min}\right) = 0,410116 + \frac{4}{5}\left(0,6829 - 0,410116\right) = 0,62835$$

Kemudian batas atas masing-masing subset dihitung sebagai berikut

$$thrUp_1 = H_{\min} + \frac{1}{m}\left(H_{\max} - H_{\min}\right) = 0,410116 + \frac{1}{5}\left(0,6829 - 0,410116\right) = 0,464675$$

$$thrUp_2 = H_{\min} + \frac{2}{m}\left(H_{\max} - H_{\min}\right) = 0,410116 + \frac{2}{5}\left(0,6829 - 0,410116\right) = 0,519233$$

$$\vdots$$

$$thrUp_5 = H_{\min} + \frac{5}{m}\left(H_{\max} - H_{\min}\right) = 0,410116 + \frac{5}{5}\left(0,6829 - 0,410116\right) = 0,6829$$

Hasil dari batas atas, batas bawah, serta anggota dan nilai *fuzzy membership* masing-masing subset ditunjukkan pada Tabel 4.4.

Tabel 4.4 Anggota tiap Subset

Subset	thrLow	thrUp	Anggota (data ke-)	FM_l
1	0.410116	0.464675	11	1
2	0.464675	0.519233	-	0.95
3	0.519233	0.573791	-	0.9
4	0.573791	0.62835	10, 12	0.85
5	0.62835	0.682908	7, 8, 9	0.8

Anggota dari subset pertama adalah unit sampel negatif dengan nilai *entropy* antara 0,410116 sapai 0,464675, begitu seterusnya hingga subset ke-5. Subset ke-2 dan ke-3 kosong karena tidak ada unit sampel dari kelas negatif yang nilai *entropy*nya berada pada range tersebut. Kolom FM_l menunjukkan nilai *fuzzy membership* untuk masing masing subset yang dihitung sebagai berikut.

$$FM_1 = 1.0 - 0.05 * (1-1) = \quad 0$$
$$FM_2 = 1.0 - 0.05 * (2-1) = 0,95$$
$$\vdots$$
$$FM_3 = 1.0 - 0.05 * (3-1) = \quad 0,8$$

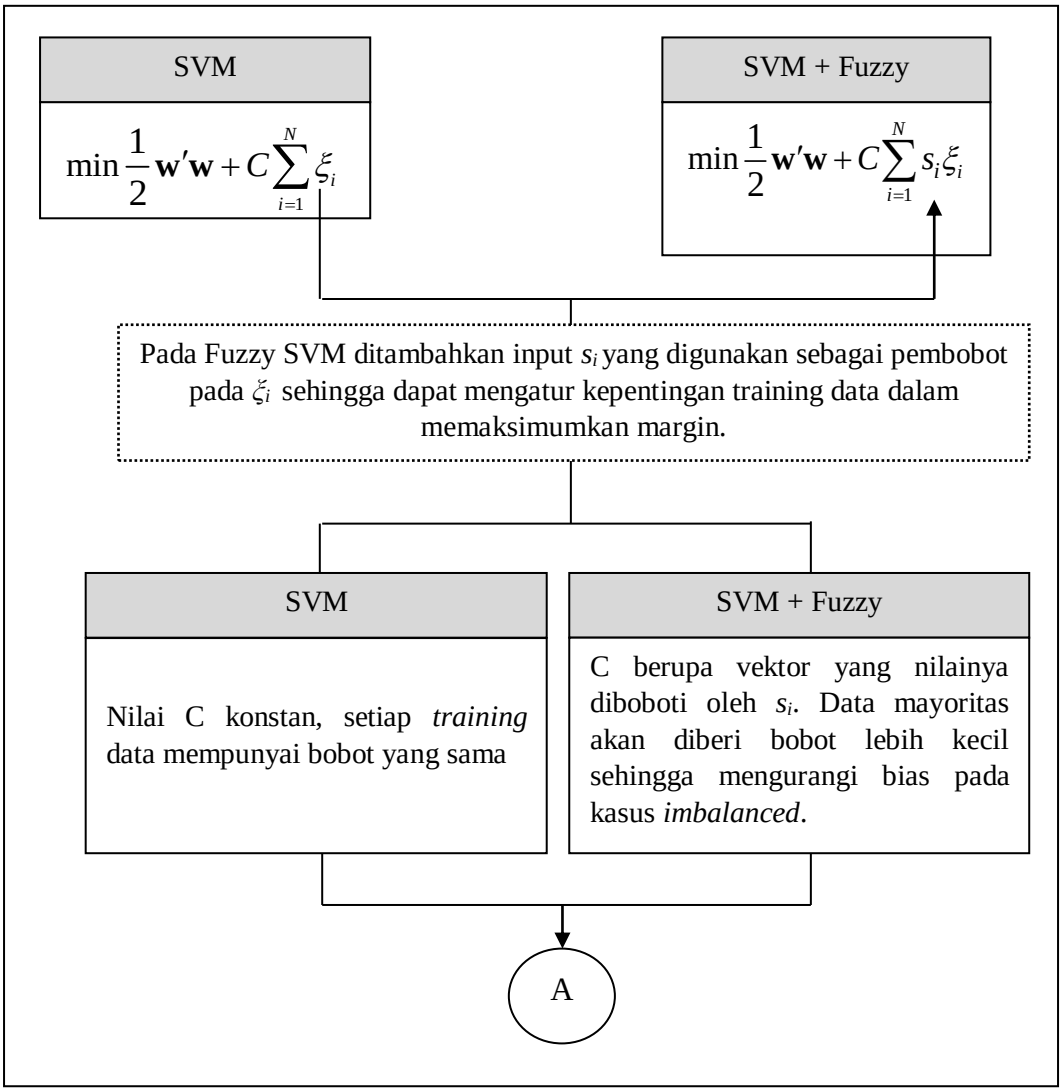
Selanjutnya berdasarkan persamaan 2.37 dengan menggunakan nilai FM_l pada Tabel 4.4 didapatkan *entropy based fuzzy membership* pada Tabel 4.5

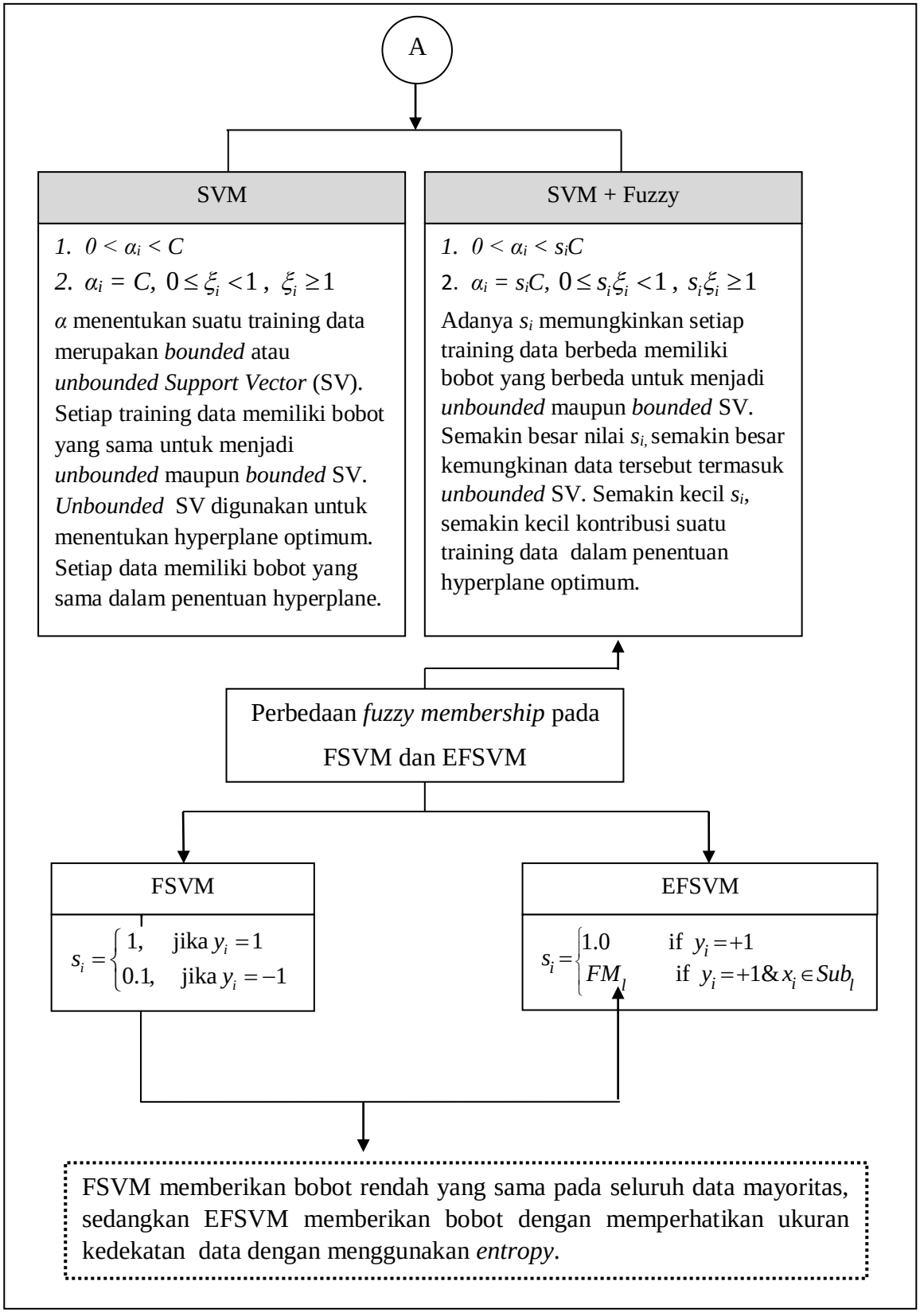
Tabel 4.5 *Entropy based Fuzzy Membership* dari Data Ilustrasi

Data	s_i
1	1
2	1
3	1
4	1
5	1
6	1
7	0.8
8	0.8
9	0.8
10	0.85
11	1
12	0.85

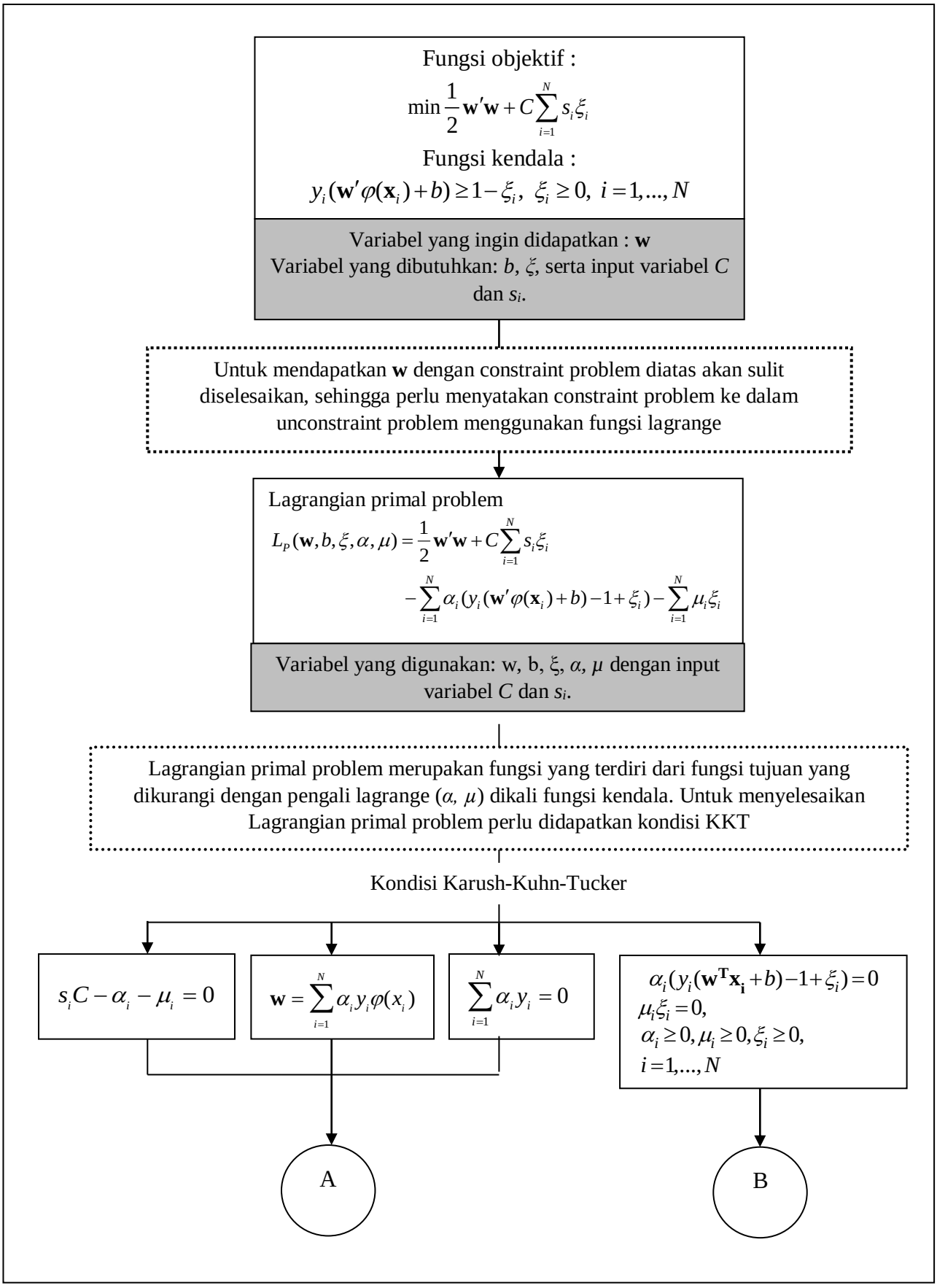
4.1.2 Penerapan *Fuzzy* pada *Support Vector Machine*

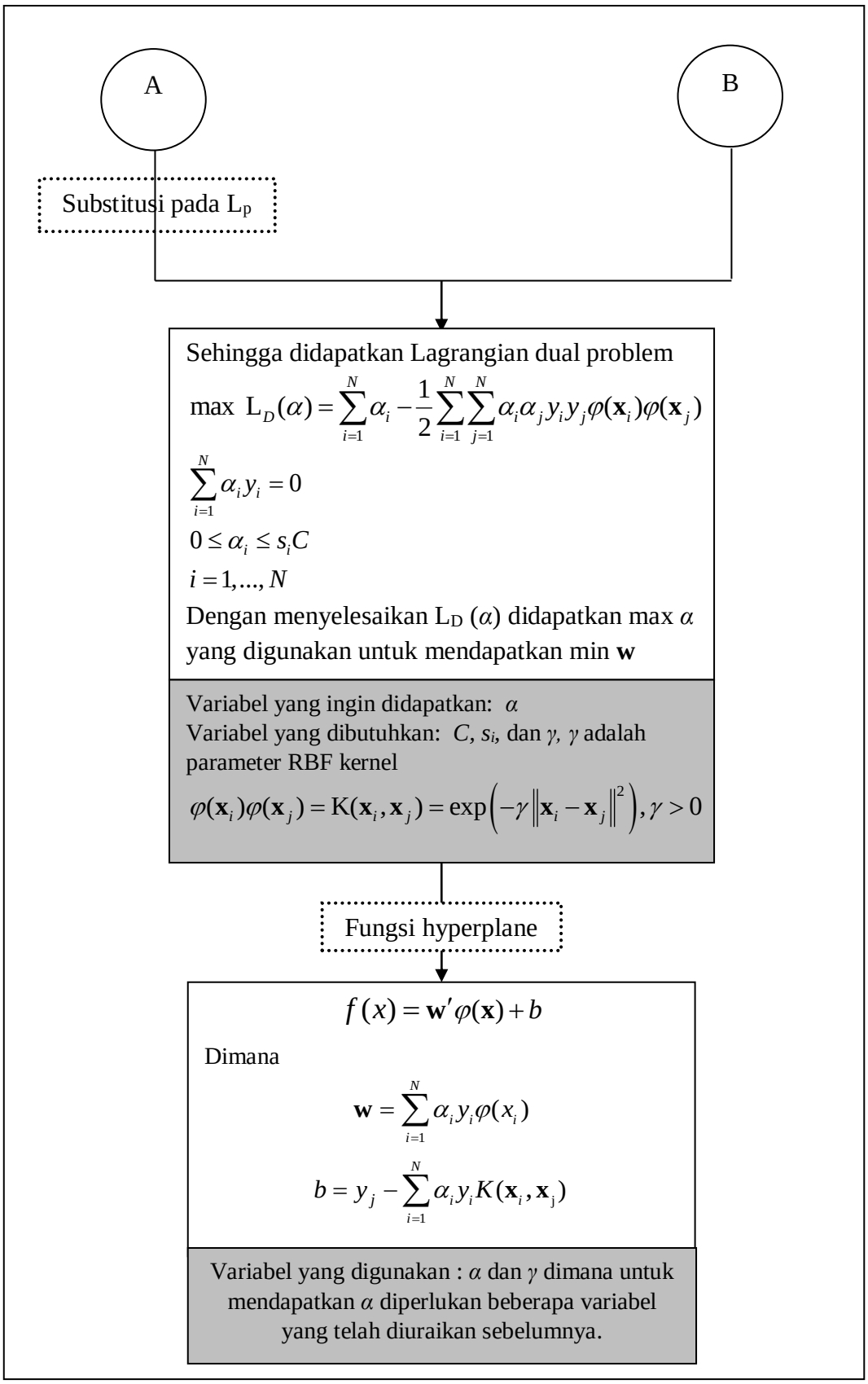
Pada bagian ini akan dijelaskan bagaimana mendapatkan *hyperplane* pemisah yang optimal pada EFSVM. *Input* yang digunakan adalah $S = \{(\mathbf{x}_i, y_i, s_i)\}_{i=1}^{n}$, dimana $\mathbf{x}_i$ adalah sampel berukuran *n*, $y_i \in \{+1, -1\}$ yang menyatakan kelas (+1 untuk kelas positif dan -1 untuk kelas negatif), dan s_i adalah keanggotaan *entropy* berdasarkan *fuzzy*. Sebelum menjelaskan bagaimana mendapatkan hyperplane pemisah, diberikan review EFSVM yang menunjukkan perbedaan EFSVM dengan SVM dan FSVM yang ditunjukkan pada Gambar 4.3 serta variabel-variabel yang digunakan dalam mendapatkan fungsi hyperplane pemisah pada Gambar 4.4.





Gambar 4.3 Perbedaan SVM, FSVM, dan EFSVM


Fungsi objektif :
$\min \frac{1}{2}\mathbf{w}'\mathbf{w} + C\sum_{i=1}^{N} s_i\xi_i$
Fungsi kendala :
$y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b) \geq 1-\xi_i,\ \xi_i \geq 0,\ i=1,...,N$
Variabel yang ingin didapatkan : **w**
Variabel yang dibutuhkan: *b*, *ξ*, serta input variabel *C* dan s_i.
Untuk mendapatkan **w** dengan constraint problem diatas akan sulit diselesaikan, sehingga perlu menyatakan constraint problem ke dalam unconstraint problem menggunakan fungsi lagrange
Lagrangian primal problem
$L_P(\mathbf{w},b,\xi,\alpha,\mu) = \frac{1}{2}\mathbf{w}'\mathbf{w} + C\sum_{i=1}^{N} s_i\xi_i - \sum_{i=1}^{N}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i) - \sum_{i=1}^{N}\mu_i\xi_i$
Variabel yang digunakan: w, b, ξ, *α*, *μ* dengan input variabel *C* dan s_i.
Lagrangian primal problem merupakan fungsi yang terdiri dari fungsi tujuan yang dikurangi dengan pengali lagrange (*α*, *μ*) dikali fungsi kendala. Untuk menyelesaikan Lagrangian primal problem perlu didapatkan kondisi KKT
Kondisi Karush-Kuhn-Tucker
$s_iC - \alpha_i - \mu_i = 0$
$\mathbf{w} = \sum_{i=1}^{N}\alpha_i y_i \varphi(x_i)$
$\sum_{i=1}^{N}\alpha_i y_i = 0$
$\alpha_i(y_i(\mathbf{w}^{\mathbf{T}}\mathbf{x}_{\mathbf{i}}+b)-1+\xi_i)=0$
$\mu_i\xi_i = 0,$
$\alpha_i \geq 0, \mu_i \geq 0, \xi_i \geq 0,$
$i=1,...,N$
A
B



Gambar 4.4 Langkah mendapatkan Hyperplane Pemisah pada EFSVM

Untuk menemukan hyperplane pemisah yang optimal maka perlu untuk menyelesaikan masalah optimasi kuadrat pada persamaan (2.39). Fungsi tujuan dan fungsi kendala terlebih dahulu dirubah kedalam bentuk fungsi lagrange pada persamaan (2.41), dimana α_i dan μ_i adalah non-negative *lagrange multiplier*. Solusi optimal memenuhi kondisi Karush–Kuhn–Tucker (KKT) yang ditunjukkan pada persamaan (2.42), (2.43), dan (2.44). Untuk turunan terhadap **w** ditunjukkan sebagai berikut.

$$
\begin{aligned}
&\frac{\partial L(\mathbf{w},b,\xi,\alpha,\mu)}{\partial w}=0\\
&\Leftrightarrow \frac{\partial\left(\frac{1}{2}\mathbf{w}'\mathbf{w}+C\sum_{i=1}^{m}s_i\xi_i-\sum_{i=1}^{m}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i)-\sum_{i=1}^{m}\mu_i\xi_i\right)}{\partial\mathbf{w}}=0\\
&\Leftrightarrow \frac{\partial\left(\frac{1}{2}\mathbf{w}'\mathbf{w}\right)}{\partial\mathbf{w}}+\frac{\partial\left(C\sum_{i=1}^{m}s_i\xi_i\right)}{\partial\mathbf{w}}+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i y_i\mathbf{w}'\varphi(\mathbf{x}_i)\right)}{\partial\mathbf{w}}\\
&+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i(y_i b-1+\xi_i))\right)}{\partial\mathbf{w}}+\frac{\partial\left(\sum_{i=1}^{m}\mu_i\xi_i\right)}{\partial\mathbf{w}}=0\\
&\Leftrightarrow\left(\frac{1}{2}\cdot 2\cdot\mathbf{w}\right)+0-\left(\sum_{i=1}^{m}\alpha_i y_i\varphi(\mathbf{x}_i)\right)-0+0=0\\
&\Leftrightarrow \mathbf{w}-\sum_{i=1}^{m}\alpha_i y_i\varphi(\mathbf{x}_i)=0\\
&\Leftrightarrow \mathbf{w}=\sum_{i=1}^{m}\alpha_i y_i\varphi(\mathbf{x}_i)
\end{aligned}
\tag{4.1}
$$

Setelah diturunkan terhadap **w** maka didapatkan hasil $\mathbf{w}=\sum_{i=1}^{m}\alpha_i y_i\varphi(\mathbf{x}_i)$. Kemudian dengan cara yang sama, fungsi *lagrange primal problem* diturunkan terhadap *b*.

$$
\begin{aligned}
&\frac{\partial L(\mathbf{w},b,\xi,\alpha,\mu)}{\partial b}=0\\
&\Leftrightarrow \frac{\partial\left(\frac{1}{2}\mathbf{w}'\mathbf{w}+C\sum_{i=1}^{m}s_i\xi_i-\sum_{i=1}^{m}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i)-\sum_{i=1}^{m}\mu_i\xi_i\right)}{\partial b}=0\\
&\Leftrightarrow \frac{\partial\left(\frac{1}{2}\mathbf{w}'\mathbf{w}\right)}{\partial b}+\frac{\partial\left(C\sum_{i=1}^{m}s_i\xi_i\right)}{\partial b}+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i)\right)}{\partial b}\\
&+\frac{\partial\left(-\sum_{i=1}^{m}\mu_i\xi_i\right)}{\partial b}=0
\end{aligned}
$$

$$\Leftrightarrow \frac{\partial\left(\frac{1}{2}\mathbf{w}'\mathbf{w}\right)}{\partial b}+\frac{\partial\left(C\sum_{i=1}^{m}s_i\xi_i\right)}{\partial b}+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i y_i\mathbf{w}'\varphi(\mathbf{x}_i)\right)}{\partial b}+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i y_i b\right)}{\partial b}$$
$$+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i(-1+\xi_i)\right)}{\partial b}+\frac{\partial\left(\sum_{i=1}^{m}\mu_i\xi_i\right)}{\partial b}=0$$
$$\Leftrightarrow 0+0-0-\sum_{i=1}^{m}\alpha_i y_i+0=0 \qquad (4.2)$$
$$\Leftrightarrow -\sum_{i=1}^{m}\alpha_i y_i=0$$

Setelah diturunkan terhadap b didapatkan hasil $\sum_{i=1}^{m}\alpha_i y_i=0$ dan yang terakhir yaitu diturunkan terhadap ξ sebagai berikut.

$$\frac{\partial L(\mathbf{w},b,\xi,\alpha,\mu)}{\partial\xi_i}=0$$
$$\Leftrightarrow \frac{\partial\left(\frac{1}{2}\mathbf{w}'\mathbf{w}+C\sum_{i=1}^{m}s_i\xi_i-\sum_{i=1}^{m}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i)-\sum_{i=1}^{m}\mu_i\xi_i\right)}{\partial\xi_i}=0$$
$$\Leftrightarrow \frac{\partial\left(\frac{1}{2}\mathbf{w}'\mathbf{w}\right)}{\partial\xi_i}+\frac{\partial\left(C\sum_{i=1}^{m}s_i\xi_i\right)}{\partial\xi_i}+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i)\right)}{\partial\xi_i}$$
$$+\frac{\partial\left(-\sum_{i=1}^{m}\mu_i\xi_i\right)}{\partial\xi_i}=0$$
$$\Leftrightarrow \frac{\partial\left(\frac{1}{2}\mathbf{w}'\mathbf{w}\right)}{\partial\xi_i}+\frac{\partial\left(C\sum_{i=1}^{m}s_i\xi_i\right)}{\partial\xi_i}+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1)\right)}{\partial\xi_i}$$
$$+\frac{\partial\left(-\sum_{i=1}^{m}\alpha_i\xi_i\right)}{\partial\xi_i}+\frac{\partial\left(-\sum_{i=1}^{m}\mu_i\xi_i\right)}{\partial\xi_i}=0$$
$$\Leftrightarrow 0+s_iC-0-\alpha_i-\mu_i=0 \qquad (4.3)$$
$$\Leftrightarrow \alpha_i+\mu_i=s_iC$$

Setelah diturunkan terhadap ξ didapatkan hasil $\alpha_i+\mu_i=s_iC$. Masing-masing kondisi KKT yang telah didapatkan yaitu persamaan (2.46), (2.47), dan (2.48), kemudian disubstitusikan pada persamaan (2.41) sebagai berikut.

$$
\begin{aligned}
L &= \frac{1}{2}\mathbf{w}'\mathbf{w} + C\sum_{i=1}^{m} s_i\xi_i - \sum_{i=1}^{m}\alpha_i(y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i) - \sum_{i=1}^{m}\mu_i\xi_i \\
&= \frac{1}{2}\left(\sum_{i=1}^{m}\alpha_i y_i\varphi(\mathbf{x}_i)\right)'\left(\sum_{i=1}^{m}\alpha_i y_i\varphi(\mathbf{x}_i)\right) + C\sum_{i=1}^{l} s_i\xi_i \\
&\quad - \sum_{i=1}^{m}\alpha_i\left(y_i\left(\left(\sum_{i=1}^{m}\alpha_i y_i\varphi(\mathbf{x}_i)\right)'\varphi(\mathbf{x}_i)+b\right)-1+\xi_i\right) - \sum_{i=1}^{m}\mu_i\xi_i \\
&= \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\varphi(\mathbf{x}_i)\varphi(\mathbf{x}_j) + \sum_{i=1}^{l} Cs_i\xi_i - \sum_{i=1}^{m}\alpha_i y_i\left(\sum_{i=1}^{m}\alpha_i y_i\varphi(\mathbf{x}_i)\right)'\varphi(\mathbf{x}_i) \\
&\quad - \sum_{i=1}^{m}\alpha_i y_i b + \sum_{i=1}^{m}\alpha_i - \sum_{i=1}^{m}\alpha_i\xi_i - \sum_{i=1}^{m}\mu_i\xi_i \\
&= \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\varphi(\mathbf{x}_i)\varphi(\mathbf{x}_j) + \sum_{i=1}^{l}(\alpha_i+\beta_i)\xi_i - \sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\varphi(\mathbf{x}_i)\varphi(\mathbf{x}_j) \\
&\quad - b\sum_{i=1}^{m}\alpha_i y_i + \sum_{i=1}^{m}\alpha_i - \sum_{i=1}^{m}\alpha_i\xi_i - \sum_{i=1}^{m}\mu_i\xi_i \\
&= \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\varphi(\mathbf{x}_i)\varphi(\mathbf{x}_j) - \sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\varphi(\mathbf{x}_i)\varphi(\mathbf{x}_j) \\
&\quad + \sum_{i=1}^{l}\alpha_i\xi_i + \sum_{i=1}^{l}\mu_i\xi_i - \sum_{i=1}^{m}\alpha_i\xi_i - \sum_{i=1}^{m}\mu_i\xi_i - b\sum_{i=1}^{m}\alpha_i y_i + \sum_{i=1}^{m}\alpha_i \\
&= -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\varphi(\mathbf{x}_i)\varphi(\mathbf{x}_j) + 0 + 0 - b\cdot 0 + \sum_{i=1}^{m}\alpha_i \qquad (4.4) \\
&= \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\varphi(\mathbf{x}_i)\varphi(\mathbf{x}_j)
\end{aligned}
$$

Hasil dari substitusi tersebut merupakan fungsi *lagrange dual problem* seperti yang ditunjukkan pada persamaan (2.49) beserta fungsi kendalanya. Persamaan (2.49) merupakan bentuk kuadratik sehingga dapat diselesaikan dengan berbagai *quadratic programming* (QP) *solver*. Pada penelitian ini digunakan ipop untuk menyelesaikan bentuk *quadratic programming*. Ipop menerapkan *interior point methods* untuk menyelesaikan *quadratic programming problem* yang bentuk umumnya dinyatakan sebagai berikut (Vanderbei, 1998).

$$\min \quad \mathbf{c}'\mathbf{x} + \frac{1}{2}\mathbf{x}'\mathbf{H}\mathbf{x} \qquad (4.5)$$

dengan fungsi kendala:

$$\mathbf{b} \le \mathbf{A}\mathbf{x} \le \mathbf{b} + \mathbf{r}\,;\ \mathbf{l} \le \mathbf{x} \le \mathbf{u} \qquad (4.6)$$

Tidak seperti QP solver lainnya yang hanya mampu menyelesaikan matriks *positive definite*, ipop mampu menyelesaikan matriks *positive semidefinite*.

Hasil dari fungsi lagrange dual problem ini adalah $\boldsymbol{\alpha}$. Dari fungsi kendala pada persamaan (2.45) serta hasil KKT pada persamaan (2.48) terdapat tiga kondisi untuk $\boldsymbol{\alpha}$ yaitu:

1. Apabila $\alpha_i = 0$ maka $s_i\xi_i = 0$ maka x_i diklasifikasikan dengan benar.
2. Saat nilai $0 < \alpha_i < s_iC$ maka $y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i = 0$ dan $s_i\xi_i = 0$. Sehingga $y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b) = 1$ dan $\mathbf{x_i}$ adalah *support vector* atau lebih tepatnya *unbounded support vector*. Jadi untuk $s_iC > \alpha_i > 0$ akan menghasilkan *unbounded support vector*. *Unbounded support vector* inilah yang nantinya menentukan *hyperplane* yang optimum.
3. Apabila $\alpha_i = s_iC$ maka $y_i(\mathbf{w}'\varphi(\mathbf{x}_i)+b)-1+\xi_i = 0$ dan $s_i\xi_i \geq 0$ Sehingga $\mathbf{x_i}$ adalah *support vector* atau lebih tepatnya *bounded support vector*. Jadi jika $0 \leq s_i\xi_i < 1$ maka $\mathbf{x_i}$ diklasifikasikan benar, namun jika $s_i\xi_i \geq 1$ maka $\mathbf{x_i}$ *misclassified*.

Perbedaan antara SVM dan SVM dengan penambahan keanggotaan fuzzy terletaak pada penerapan fuzzy membership (s_i). Nilai s_i yang lebih kecil mengurangi efek dari parameter ξ_i pada persamaan (2.39) sehingga sampel $\mathbf{x_i}$ dikurangi kepentingannya. Selain itu berdasarkan kondisi pada α_i maka titik dengan nilai α_i yang sama kemungkinan bisa memiliki tipe support vector yang berbeda dikarenakan adanya s_i. Kemudian nilai α_i yang telah didapatkan digunakan untuk mendapatkan fungsi keputusan pada persamaan (2.52). Persamaan (2.52) didapatkan dengan mensubstitusikan persamaan (2.51) kedalam hyperplane pemisah sebagai berikut.

$$\begin{aligned} f(x) &= \mathbf{w}'\varphi(\mathbf{x}) + b \\ &= \left(\sum_{i=1}^{m} \alpha_i y_i \varphi(\mathbf{x}_i) \right)' \varphi(\mathbf{x}) + b \\ &= \sum_{i=1}^{m} \alpha_i y_i \varphi(\mathbf{x}_i)' \varphi(\mathbf{x}) + b \end{aligned} \tag{4.8}$$

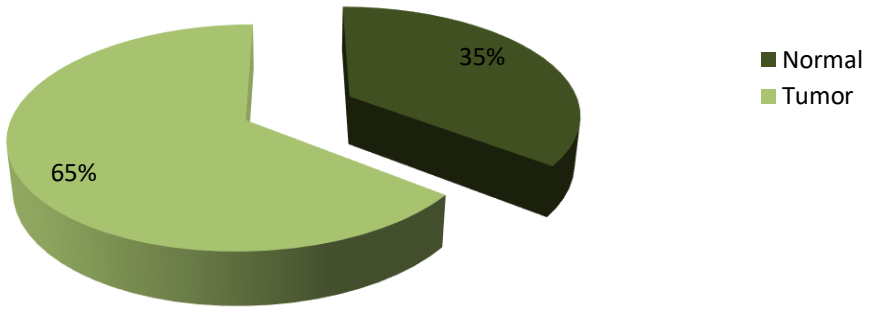
Sehingga didapatkan *decision function* pada persamaan (2.52).

4.2 Karakteristik Data DNA Microarray

Bagian ini memuat karakteristik dari data colon cancer dan breast cancer. Setiap data tersebut memiliki karakteristik yang berbeda dilihat dari pola persebaran data dari setiap atribut-atribut dan kategori kelasnya. Berikut merupakan karakteristik dari masing-masing data.

a. Data *Colon Cancer*

Data *colon cancer* diperoleh dari pengamatan yang dilakukan pada jaringan usus manusia. Dari 62 sampel yang diambil, data tersebut dibagi ke dalam dua kelas yaitu kelas tumor dan kelas normal. Deskripsi dari kedua kelas ini ditunjukkan pada Gambar 4.5.

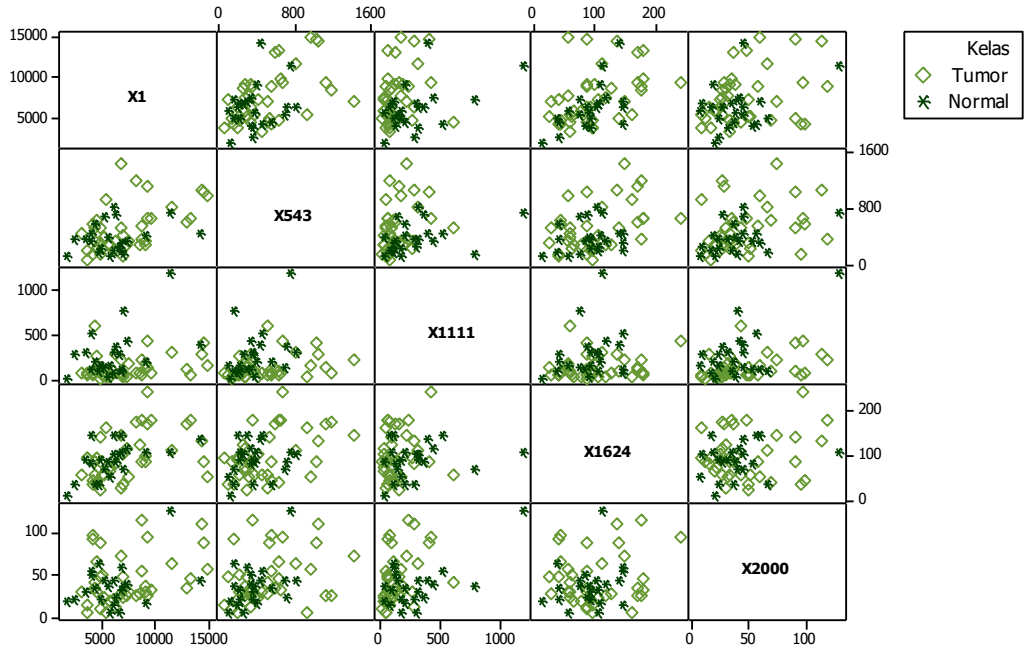


Gambar 4.5 Persentase Kelas Normal dan Tumor pada Data *Colon Cancer*

Gambar 4.5 menunjukkan bahwa data *colon cancer* merupakan data yang *imbalanced* dilihat dari perbandingan proporsi kelas tumor dan normal yang besar. Pada Gambar 4.5 terlihat bahwa dari 62 sampel, 35% diantaranya yaitu 22 sampel dinyatakan normal sedangkan sisanya sebanyak 40 sampel atau sekitar 65% dari seluruh sampel adalah tumor.

Sebanyak 2000 gen digunakan sebagai penilaian untuk menentukan sampel termasuk ke dalam kelas tumor maupun normal. Karena ukuran jumlah gen yang digunakan sangat besar maka dapat dipastikan bahwa pola persebaran data menjadi sangat kompleks. Gambar 4.6 menunjukkan pola sebaran data, dari beberapa *feature* untuk *colon cancer dataset* pola sebaran datanya sangat kompleks. Pola data terlihat tersebar merata sehingga menyulitkan klasifikasi. Selain itu, dari pola data menunjukkan bahwa fungsi pemisah dari klasifikasinya

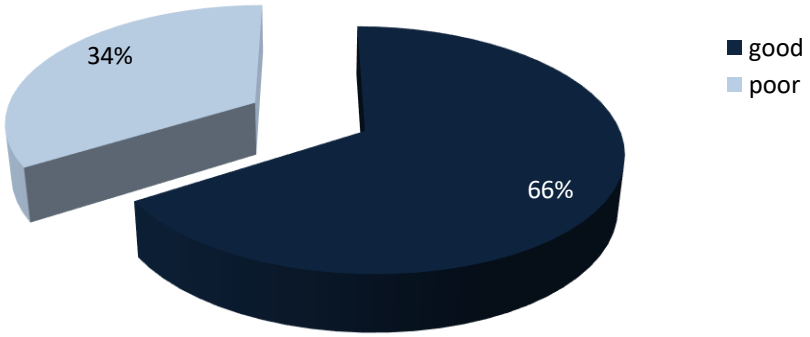
akan berupa nonlinier sehingga akan digunakan fungsi kernel untuk mempermudah klasifikasi data *colon cancer*.



Gambar 4.6 Persebaran Data dari Beberapa *Feature* pada Data *Colon Cancer*

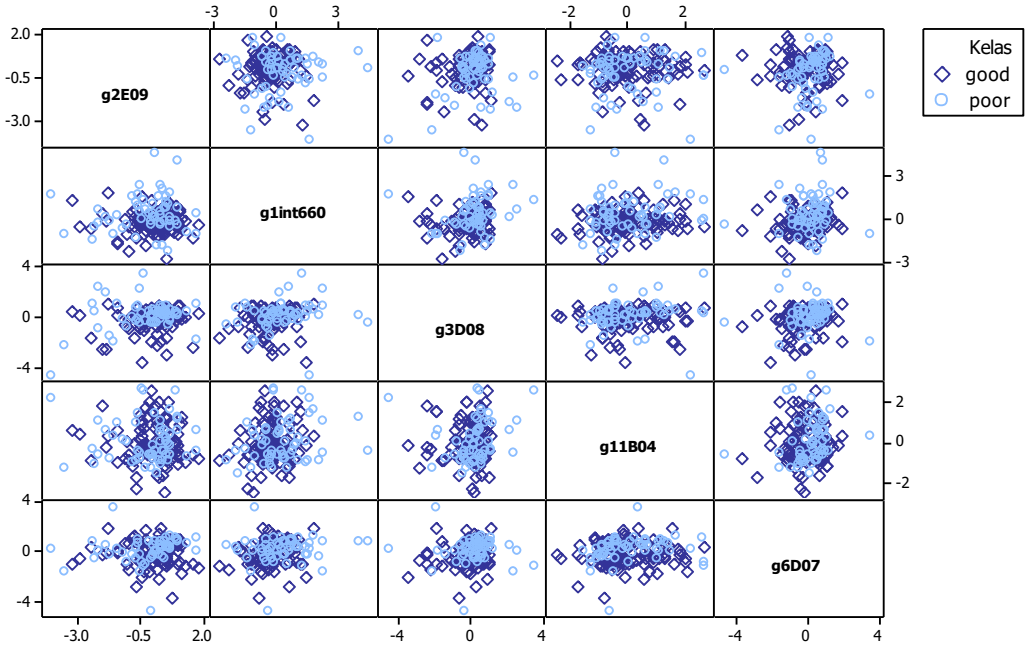
b. Data *Breast Cancer*

Data *breast cancer* didapatkan dari pengamatan yang dilakukan terhadap 168 sampel. Hasil dari pengamatan tersebut, 168 sampel terbagi kedalam dua kelas yaitu kelas "*poor"* dan kelas "*good"*. Deskripsi dari kedua kelas tersebut ditunjukkan pada Gambar 4.7.



Gambar 4.7 Persentase Kelas "*Good*" dan "*Poor*" pada Data *Breast Cancer*

Sama halnya dengan data *colon cancer*, Gambar 4.7 menunjukkan bahwa data *breast cancer* juga merupakan data yang *imbalanced* dilihat dari perbandingan proporsi kelas "*poor"* dan "*good"* yang besar. Pada Gambar 4.7 terlihat bahwa dari 168 sampel, 111 diantaranya atau sekitar 66% sampel tergolong dalam kelas "*good"* sedangkan sisanya yaitu 57 sampel atau sekitar 44% dari seluruh sampel termasuk ke dalam kelas "*poor*". Data *breast cancer* terdiri dari 2905 gen yang digunakan sebagai penilaian untuk menentukan pasien termasuk ke dalam kelas "*good"* maupun "*poor*". Banyaknya jumlah gen pada data *breast cancer* membuat pola persebaran data menjadi sangat kompleks. Pola persebaran data breast cancer pada beberapa variabel ditunjukkan pada Gambar 4.8.



Gambar 4.8 Persebaran Data dari Beberapa *Feature* pada Data *Breast Cancer*

Gambar 4.8 menjelaskan bahwa persebaran pola data dari beberapa *feature* pada *breast cancer dataset* sangat kompleks dan tersebar secara acak sehingga menyulitkan proses klasifikasi. Pola data juga tidak menunjukkan fungsi pemisah yang linear sehingga pada analisis selanjutnya akan digunakan bantuan kernel untuk menentukan fungsi pemisah pada data *breast cancer*.

4.3 *Feature Selection*

DNA microarray merupakan *highdimensional* data yaitu data dengan jumlah *feature* yang sangat besar namun ukuran sampel kecil, sehingga perlu dilakukan *feature selection* terlebih dahulu. *Feature selection* diharapkan mampu meningkatkan performansi klasifikasi serta mempercepat proses komputasi. Berikut merupakan ringkasan hasil *feature selection* dengan menggunakan metode FCBF (*threshold* = 0) untuk kedua *dataset*.

Tabel 4.6 Hasil *Feature Selection*

Dataset	Jumlah seluruh *feature*	Hasil *feature selection*
Colon Cancer	2000	15
Breast Cancer	2905	51

Tabel 4.6 menunjukkan hasil *feature selection* yang diperoleh dengan menggunakan metode FCBF dari kedua *dataset*. Pada data *colon tumor* dari jumlah *feature* sebanyak 2000, metode FCBF mampu memilih *feature* yang relevan sebanyak 15 *feature*. Sedangkan pada data *breast cancer* dari jumlah *feature* sebanyak 7129 terpilih *feature* yang relevan sebanyak 51 *feature*. *Feature-feature* terpilih dari data *colon cancer* dan *breast cancer* masing-masing ditunjukkan pada Tabel 4.7 dan Tabel 4.8. Selanjutnya analisis dilanjutkan dengan menggunakan semua *feature* dan analisis dengan hanya menggunakan *feature* terpilih pada masing-masing metode SVM, FSVM, dan EFSVM, untuk kemudian dibandingkan performansi klasifikasinya.

Tabel 4.7 *Feature* Terpilih Data *Colon Cancer*

No	*Feature*	Information Gain	*Feature* ke-
1	X1671	0.3015691	1671
2	X249	0.2664472	249
3	X1772	0.2313463	1772
4	X625	0.2215022	625
5	X1042	0.2195625	1042
6	X1227	0.1699482	1227
7	X1153	0.1698471	1153
⋮	⋮	⋮	⋮
15	X1560	0.1067025	1560

Tabel 4.8 *Feature* Terpilih Data *Breast Cancer*

No	*Feature*	Information Gain	*Feature* ke-
1	g1CNS507	0.14777747	1337
2	g1CNS508	0.13046331	611
3	g4F12	0.10629320	1390
4	g1CNS26	0.09381501	1348
5	g1int1130	0.09178125	1875
6	g1int393	0.08205550	744
7	g1int492	0.07855726	891
8	g8D02	0.07716856	1706
9	g3F01	0.07550487	120
10	g1int1671	0.07538501	2673
11	g1int456	0.07535083	833
12	g1int1702	0.07349375	2711
13	g1int659	0.07068787	1105
⋮	⋮	⋮	⋮
51	G10C07	0.04682782	2361

4.4 Keanggotaan *Entropy based Fuzzy*

Langkah pertama dalam klasifikasi dengan EFSVM adalah menentukan keanggotaan *entropy* based *fuzzy*. Pada bagian ini akan dijelaskan bagaimana mendapatkan keanggotaan *entropy* based *fuzzy* untuk masing-masing sampel menggunakan data breast cancer pada salah satu training data yang telah dibagi menggunakan 5-*fold cross-validation*. Keanggotaan *entropy* based *fuzzy* ini nantinya akan digunakan sebagai input dalam klasifikasi dengan EFSVM yang berguna untuk menjamin kepentingan dari kelas positif (minoritas) dan mengurangi adanya bias yang disebabkan oleh kelas negatif (mayoritas). Pada keanggotaan *entropy* based *fuzzy* terlebih dahulu didapatkan nilai *entropy* dari masing-masing sampel. Nilai *entropy* ditentukan berdasarkan kedekatan satu sampel dengan yang lain menggunakan k-nearest neighbors dengan k=7. Tabel 4.9 menunjukkan matriks jarak yang digunakan untuk mendapatkan *7 nearest neighbors.*

Tabel 4.9 Matriks Jarak Sample x_i

	1	2	3	...	133
1	0	78,95	68,42	...	81,98
2	78,95	0	80,53	...	85,14
3	68,42	80,53	0	...	84,91
4	63,86	71,15	63,63	...	79,74
5	56,61	61,71	57,74	...	73,48
...	...	...	...	...	...
133	81,98	85,14	84,91	...	0

Setelah didapatkan matriks jarak, kemudian didapatkan 7 sampel dengan jarak terdekat dari masing-masing sampel. Kemudian dengan menggunakan persamaan (2.34) dan (2.35) dihitung peluang sampel tersebut masuk pada kelas positif dan negatif sehingga dari hasil tersebut dapat digunakan untuk menghitung nilai *entropy* untuk masing-masing sampel menggunakan Persamaan (2.33). Tabel 4.10 menunjukkan nilai *entropy* untuk masing-masing sampel.

Tabel 4.10 Nilai *Entropy* untuk Masing-masing Sampel

Data ke-	*Entropy* (H_i)
1	0
2	0
3	0
...	...
65	0,41011
66	0,41011
...	...
131	0,59827
132	0,41011
133	0,41011

Nilai *entropy* yang didapatkan pada Tabel 4.10 digunakan untuk mengelompokkan data pada kelas negatif kedalam subset-subset. Sampel dalam satu subset akan memiliki nilai *fuzzy* yang sama. Dengan menggunakan algoritma pada Gambar 2.6 didapatkan nilai batas atas dan batas bawah untuk masing-masing subset sebagai berikut.

Tabel 4.11 Nilai Batas atas dan Batas Bawah Subset

Subset ke-	Batas bawah	Batas atas
1	0	0.136582
2	0.136582	0.273163
3	0.273163	0.409745
4	0.409745	0.546327
5	0.546327	0.682901

Sampel pada kelas negatif dengan nilai *entropy* antara 0 sampai 0.14 akan masuk pada subset 1, selanjutnya data dengan nilai *entropy* lebih tinggi yaitu 0.136582 sampai 0.27 akan masuk pada subset 2 dan seterusnya sampai subset ke-5. Distribusi dari sampel kelas negatif untuk masing-masing subset ditunjukkan pada Tabel 4.7.

Tabel 4.12 Distribusi Sampel Kelas Negatif Pada tiap Subset

Subset ke-	Anggota kelas negatif
1	1,2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14, 16, 17, 19, 20, 21, 22, 23, 24, 26, 30, 31, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 68, 69 , 70, 71, 72, 73, 74 , 75, 77, 78, 79, 80 , 82 , 85 , 87,...,120, 122, 123, 124
2	-
3	-
4	13, 18, 25, 27, 28, 29, 76, 81, 83, 84, 86, 108, 113, 114, 121
5	15, 32

Berdasarkan Tabel 4.12 dapat dilihat bahwa untuk subset ke 2 dan 3 tidak terdapat sampel dari kelas negatif didalamnya sedangkan untuk subset ke-1 terdapat 71 sampel negatif, subset ke-4 terdapat 15 sampel dari kelas negatif, dan pada subset ke-5 terdapat 2 sampel dari kelas negatif. Kemudian keanggotaan *fuzzy* untuk sampel (FM_l) pada masing-masing subset didapatkan melalui persamaan (2.36). Nilai FM_l untuk masing-masing subset ditunjukkan pada Tabel 4.13.

Tabel 4.13 Nilai FM_l untuk Masing-masing Subset

Subset ke-	FM_l
1	1
2	0.95
3	0.9
4	0.85
5	0.8

Sehingga dengan menggunakan Persamaan (2.37) didapatkan keanggotaan *entropy based fuzzy* untuk masing-masing sampel pada Tabel 4.14.

Tabel 4.14 Keanggotaan *Entropy Based Fuzzy*

Data ke-	s_i	Data ke-	s_i
1	1	15	0.8
2	1	16	1
3	1	17	1
...	...	18	0.86
13	0,85	...	...
14	1	133	1

Nilai s_i pada Tabel 4.14 selanjutnya digunakan sebagai input dalam klasifikasi menggunakan EFSVM.

4.5 Klasifikasi dengan SVM

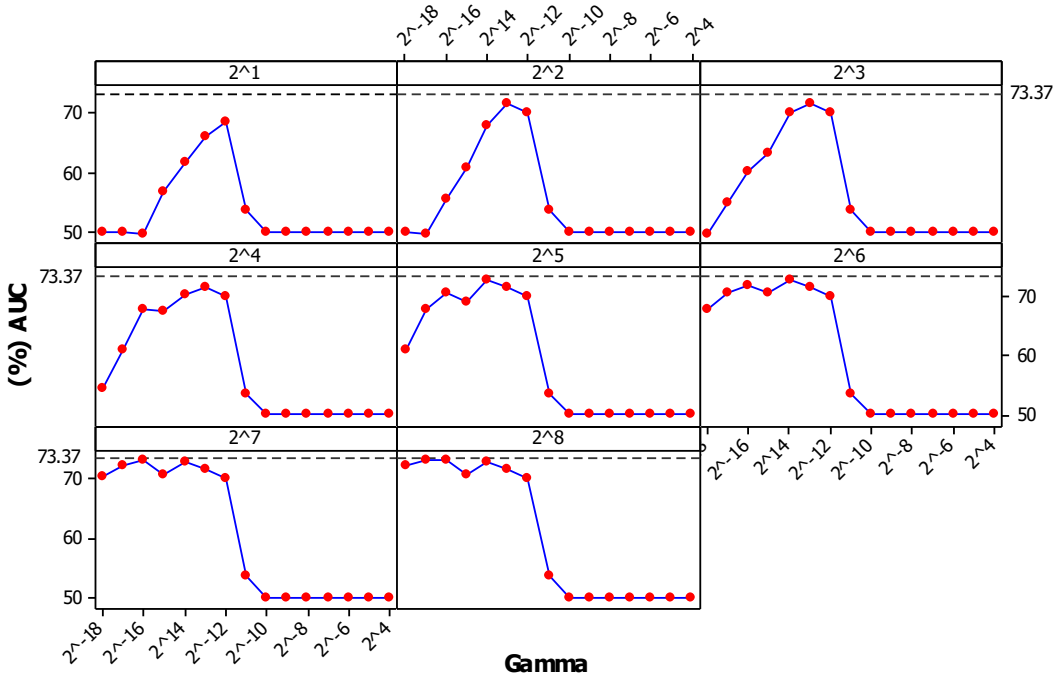
Setelah dilakukan *feature selection* kemudian dilanjutkan dengan klasifikasi pada kedua dataset. Klasifikasi yang pertama adalah klasifikasi dengan metode SVM. Parameter yang akan dioptimasi adalah parameter cost (*C*) dan gamma (γ). Parameter optimal ditentukan berdasarkan nilai rata-rata AUC (%) tertinggi yang diperoleh dari 5-*fold cross-validation* dengan kombinasi nilai parameter yang digunakan yaitu *C* pada *range* 2^1, 2^2, 2^3, 2^4, ..., 2^{11} serta nilai parameter γ pada *range* 2^{-18}, 2^{-17}, 2^{-16}, 2^{-15},..., 2^{-4}.

4.5.1 Klasifikasi Data *Breast Cancer* dengan SVM

Klasifikasi data *breast cancer* dilakukan pada data yang menggunakan seluruh *feature* dan data dengan *selected feature*. Hasil klasifikasi dinyatakan dengan nilai rata-rata persentase AUC yang didapatkan dari data testing pada 5-fold *cross-validation*.

1.)Klasifikasi menggunakan seluruh *feature*

Hasil klasifikasi pada data *breast cancer* dengan seluruh *feature* ditunjukkan pada Gambar 4.9.



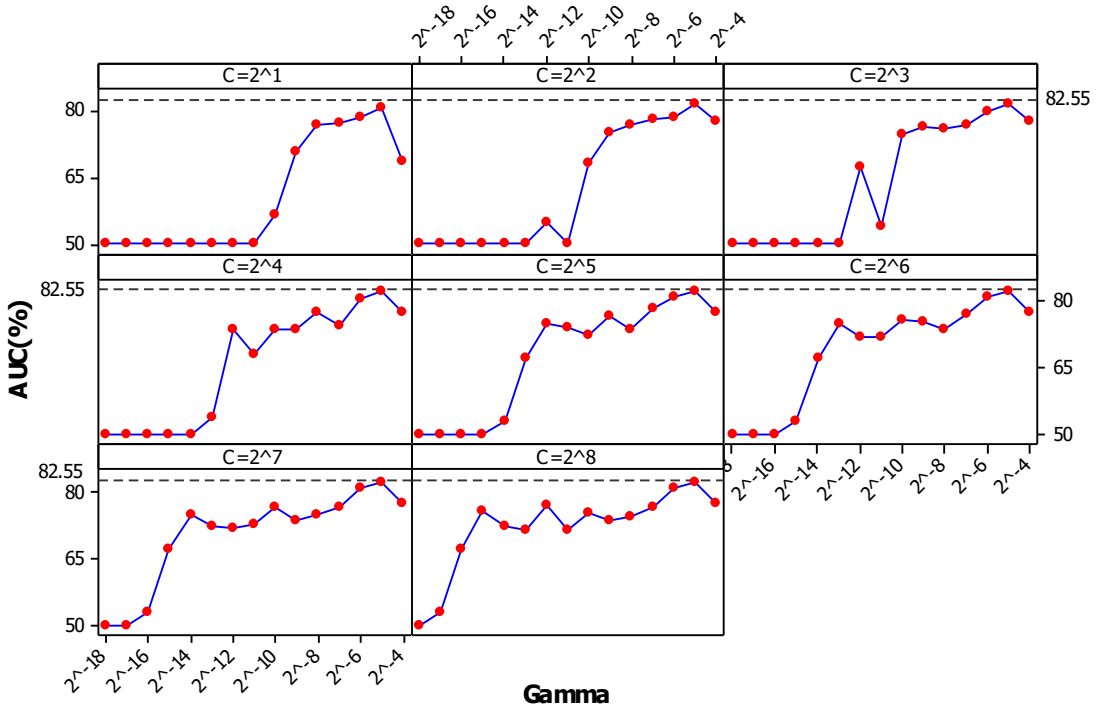
Gambar 4.9 Hasil Klasifikasi *Breast Cancer* Seluruh *Feature* dengan SVM

Berdasarkan Gambar 4.9 dapat dilihat bahwa nilai C yang lebih besar menghasilkan AUC yang tinggi. AUC terus meningkat dari nilai $C=2^1$ sampai $C=2^6$ selanjutnya yaitu $C = 2^7$ dan $C = 2^8$ nilai AUC cenderung sama. Nilai AUC sudah mencapai maksimum saat $C=2^7$ yaitu 73,3%. Adapun nilai parameter γ, saat γ sama dengan 2^{-18} sampai 2^{-12} menghasilkan nilai AUC yang tinggi namun nilainya mulai mengalami penurunan yang sangat besar saat $\gamma = 2^{-11}$. Sehingga untuk data *breast cancer* pada seluruh *feature* dengan menggunakan metode SVM didapatkan parameter yang optimal yaitu $C = 2^7$ dengan $\gamma = 2^{-16}$ dan $C = 2^7$ dengan $\gamma = 2^{-17}$ dan $\gamma = 2^{-16}$. Nilai rata-rata AUC tertinggi yang mampu dicapai adalah 73,3%. Fungsi pemisah yang terbentuk untuk klasifikasi pada Data *breast cancer* pada seluruh *feature* menggunakan SVM adalah

$$f(\mathbf{x}) = \sum_{i=1}^{133}\sum_{j=1}^{133}\alpha_i y_i \exp\left(-2^{-16}\left\|x_i - x_j\right\|^2\right) + b$$

2.)Klasifikasi menggunakan *selected feature*

Selain dilakukan klasifikasi menggunakan seluruh *feature* pada data *breast cancer*. Berikut hasil klasifikasi data breast cancer pada *selected feature*.



Gambar 4.10 Hasil Klasifikasi *Breast Cancer Selected Feature* dengan SVM

Berdasarkan Gambar 4.10 dapat dilihat bahwa nilai C yang lebih besar menghasilkan AUC yang tinggi. Nilai AUC maksimum yang mampu dicapai adalah 82,55%. Saat C = 2^1 , 2^2 , dan 2^3 AUC belum mencapai nilai maksimum. AUC mencapai nilai maksimum saat C=2^4 dan selanjutnya yaitu C=2^5 sampai C=2^8. Adapun nilai parameter γ, berbeda dengan data *breast cancer* dengan seluruh *feature*, pada klasifikasi dengan *selected feature* nilai AUC justru semakin tinggi saat saat γ semakin besar, namun nilainya menurun pada γ = 2^{-4}. Sehingga untuk data *breast cancer* pada seluruh *feature* dengan menggunakan metode SVM didapatkan parameter yang optimal yaitu C = 2^4, C = 2^5, C = 2^6, C = 2^7 dan C = 2^8 dengan γ = 2^{-5}. Fungsi pemisah yang terbentuk untuk klasifikasi pada data *breast cancer selected feature* menggunakan SVM adalah

$$f(\mathbf{x}) = \sum_{i=1}^{133}\sum_{j=1}^{133} \alpha_i y_i \exp\left(-2^{-5}\left\|x_i - x_j\right\|^2\right) + b$$

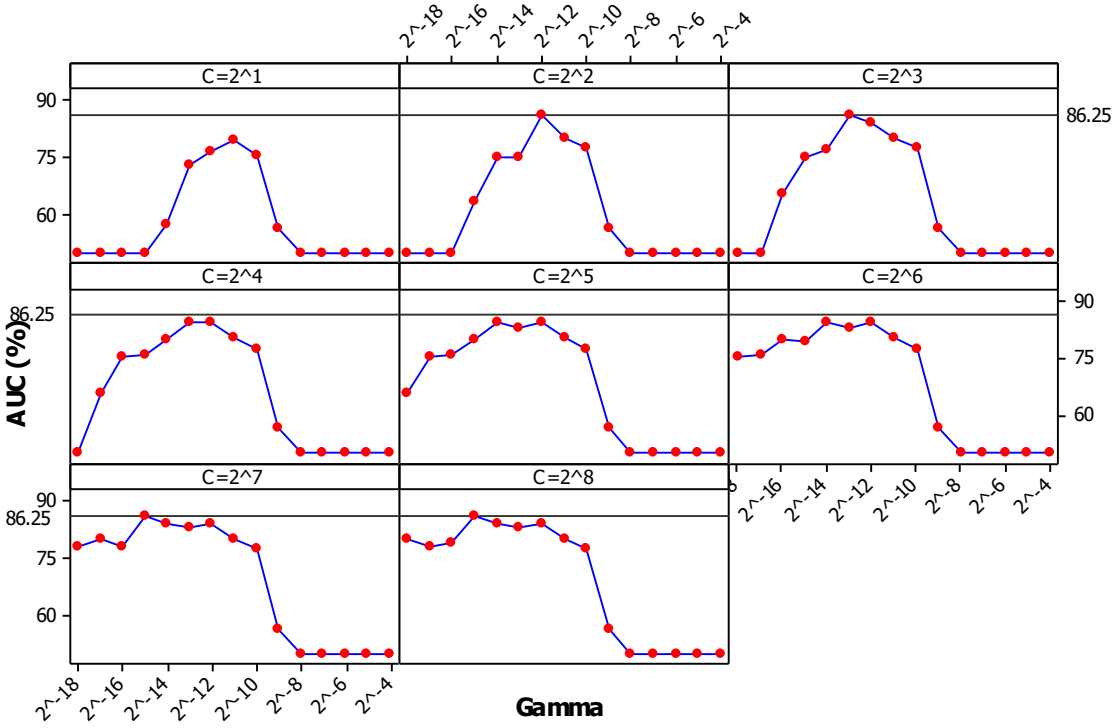
4.5.2 Klasifikasi Data *Colon Cancer* dengan SVM

Klasifikasi selanjutnya dilakukan pada data *colon cancer* dengan menggunakan seluruh *feature* dan hanya *selected feature*. Hasil klasifikasi

dinyatakan dengan nilai rata-rata persentase AUC yang didapatkan dari data testing pada 5-fold *cross-validation*.

1.) Klasifikasi menggunakan seluruh *feature*

Hasil klasifikasi pada data *colon cancer* dengan seluruh *feature* ditunjukkan pada Gambar 4.11.



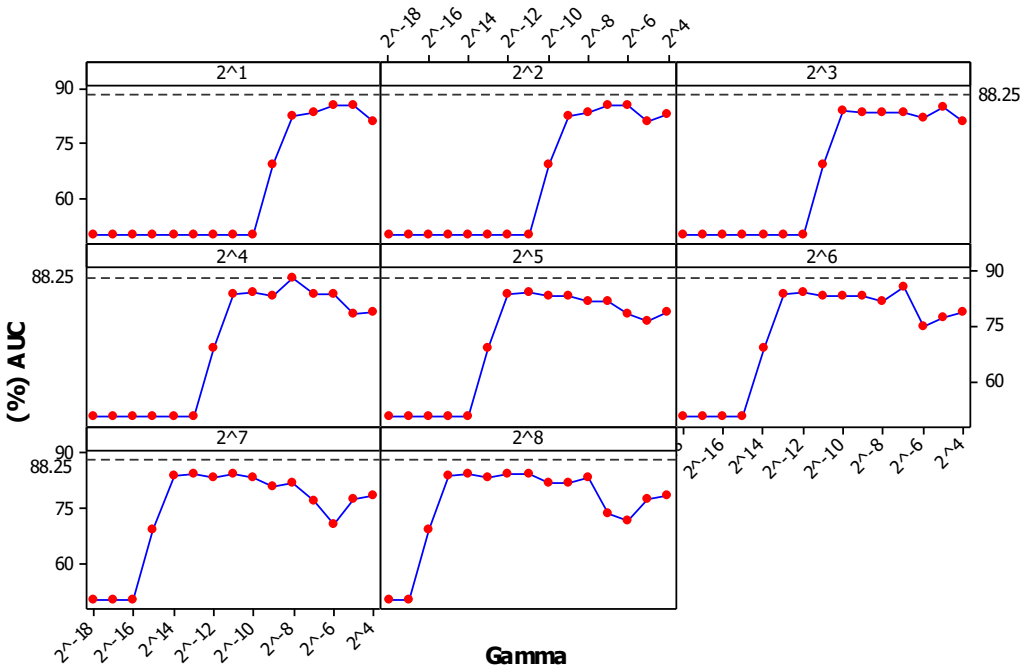
Gambar 4.11 Hasil Klasifikasi *Colon Cancer* Seluruh *Feature* dengan SVM

Berdasarkan Gambar 4.11 dapat dilihat bahwa nilai AUC maksimum yang mampu dicapai adalah 86,25% dan nilai tersebut sudah dicapai saat $C=2^2$. Adapun nilai parameter γ, saat γ sama dengan 2^{-18} sampai 2^{-15} menghasilkan nilai AUC yang rendah pada nilai parameter C yang kecil (C = 2^1, 2^2, 2^3, dan 2^4), namun nilainya meningkat pada nilai parameter C yang besar (C = 2^5, 2^6, 2^7, dan 2^8). Sedangkan untuk γ = 2^{-9}, 2^{-8},..., 2^{-4} menghasilkan nilai AUC yang kecil pada seluruh parameter C. Sehingga untuk data colon cancer pada seluruh *feature* dengan menggunakan metode SVM didapatkan parameter yang optimal yaitu C = 2^2 dengan γ = 2^{-12}, C = 2^3 dengan γ = 2^{-8}, C = 2^2 dan C = 2^2 dengan γ = 2^{-15}. Fungsi pemisah yang terbentuk untuk klasifikasi pada data *colon cancer* pada seluruh *feature* menggunakan SVM adalah

$$f(\mathbf{x}) = \sum_{i=1}^{53}\sum_{j=1}^{53} \alpha_i y_i \exp\left(-2^{-12}\left\|x_i - x_j\right\|^2\right) + b$$

2.) Klasifikasi menggunakan *selected feature*

Selain dilakukan klasifikasi menggunakan seluruh *feature* pada data *colon cancer* berikut diberikan hasil klasifikasi data *colon cancer* pada *selected feature* menggunakan SVM.



Gambar 4.12 Hasil Klasifikasi *Colon Cancer* Selected *Feature* dengan SVM

Berdasarkan Gambar 4.8 dapat dilihat bahwa nilai AUC maksimum yang mampu dicapai adalah 88,25% dan nilai tersebut sudah dicapai saat C=2^4. Adapun nilai parameter γ saat C = 2^4 mempunyai nilai yang tinggi mulai dari 2^{-11} sampai 2^{-4} dan mencapai nilai maksimum saat γ = 2^{-8}. Berbeda dengan klasifikasi sebelumnya, pada data colon cancer dengan *selected feature*, nilai maksimum hanya tercapai pada saat C = 2^4. Sehingga untuk data colon cancer pada *selected feature* dengan menggunakan metode SVM didapatkan parameter yang optimal yaitu C = 2^4 dan γ = 2^{-8}. Fungsi pemisah yang terbentuk untuk klasifikasi pada data *colon cancer* pada *selected feature* menggunakan SVM adalah

$$f(\mathbf{x}) = \sum_{i=1}^{53}\sum_{j=1}^{53} \alpha_i y_i \exp\left(-2^{-8}\left\|x_i - x_j\right\|^2\right) + b$$

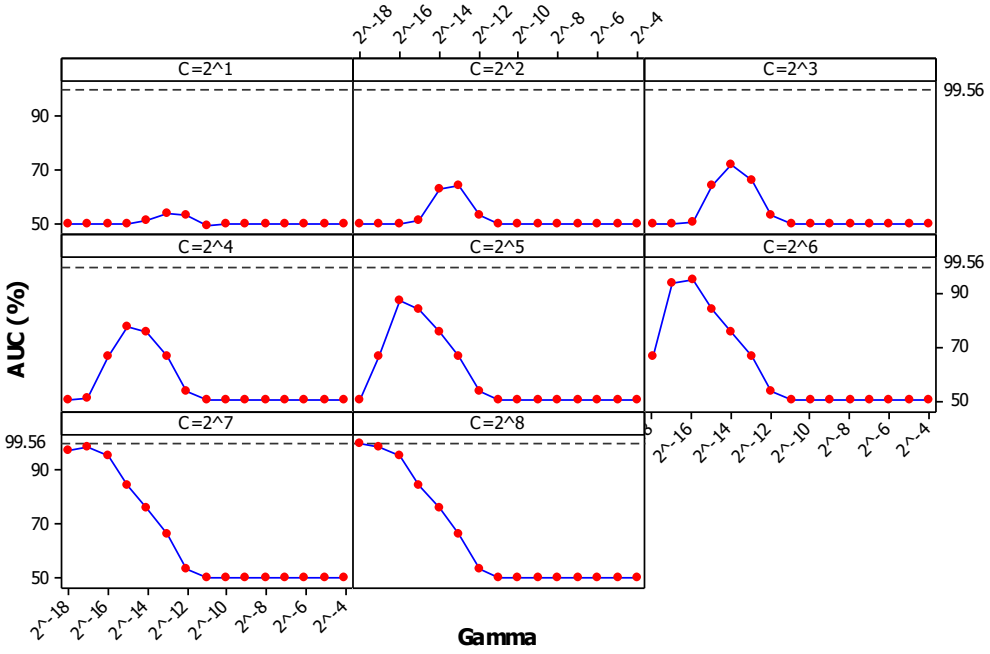
4.6 Klasifikasi Data dengan FSVM

Pada klasifikasi dengan FSVM keanggotaan *fuzzy* pada persamaan (2.32) didapatkan terlebih dahulu. Keanggotaan *fuzzy* tersebut digunakan sebagai input pada klasifikasi FSVM dimana nilainya dikalikan dengan parameter cost (C). Sama seperti klasifikasi menggunakan SVM, pada FSVM parameter yang akan dioptimasi adalah parameter cost (*C*) dan gamma (*γ)*. Parameter optimal ditentukan berdasarkan nilai rata-rata AUC tertinggi yang diperoleh dari 5-*fold cross-validation* dengan kombinasi nilai parameter yang digunakan yaitu *C* pada *range* 2^1, 2^2, 2^3, 2^4, ..., 2^{11} serta nilai parameter γ pada *range* 2^{-18}, 2^{-17}, 2^{-16}, 2^{-15},..., 2^{-4}.

4.6.1 Klasifikasi Data *Breast Cancer* dengan FSVM

1.) Klasifikasi menggunakan seluruh *feature*

Gambar 4.9 menunjukkan hasil klasifikasi data *breast cancer* dengan FSVM pada beberapa kombinasi parameter C dan γ.



Gambar 4.13 Hasil Klasifikasi *Breast Cancer* Seluruh *Feature* dengan FSVM

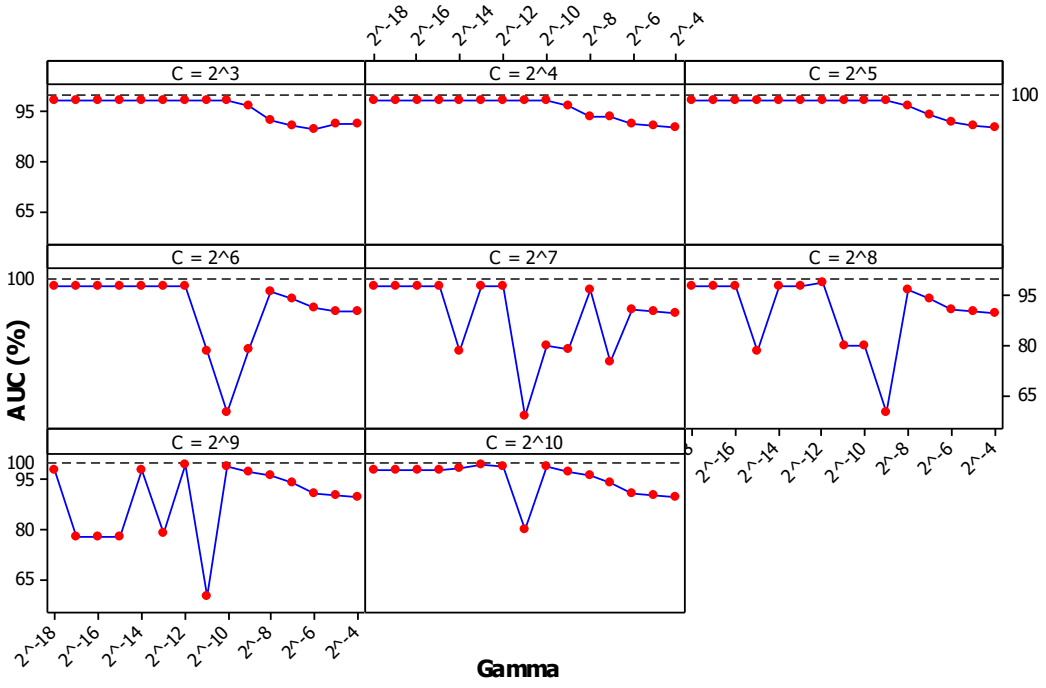
Berdasarkan Gambar 4.13 dapat dilihat bahwa nilai C yang lebih besar menghasilkan AUC yang lebih tinggi. Nilai AUC sudah mencapai maksimum saat $C=2^8$ yaitu 99,56%. Adapun nilai parameter γ, saat γ sama dengan 2^{-18} sampai 2^{-13} menghasilkan nilai AUC yang berbeda-beda pada nilai C yang berbeda, namun nilainya mulai mengalami penurunan yang besar saat $\gamma = 2^{-12}$ dan menghasilkan

nilai AUC yang kecil sampai $\gamma = 2^{-4}$. Sehingga untuk data *breast cancer* pada seluruh *feature* dengan menggunakan metode FSVM didapatkan parameter yang optimal yaitu $C = 2^{8}$ dan $\gamma = 2^{-18}$ dengan nilai rata-rata AUC tertinggi yang mampu dicapai adalah 95,56%. Fungsi pemisah yang terbentuk untuk klasifikasi pada Data *breast cancer* pada seluruh *feature* menggunakan FSVM adalah

$$f(\mathbf{x}) = \sum_{i=1}^{133}\sum_{j=1}^{133} \alpha_i y_i \exp\left(-2^{-18}\left\|x_i - x_j\right\|^2\right) + b$$

2.) Klasifikasi menggunakan *selected feature*

Selain dilakukan klasifikasi menggunakan seluruh *feature* pada data *breast cancer* berikut diberikan hasil klasifikasi data *breast cancer* pada *selected feature* menggunakan FSVM.



Gambar 4.14 Hasil Klasifikasi *Breast Cancer Selected Feature* dengan FSVM

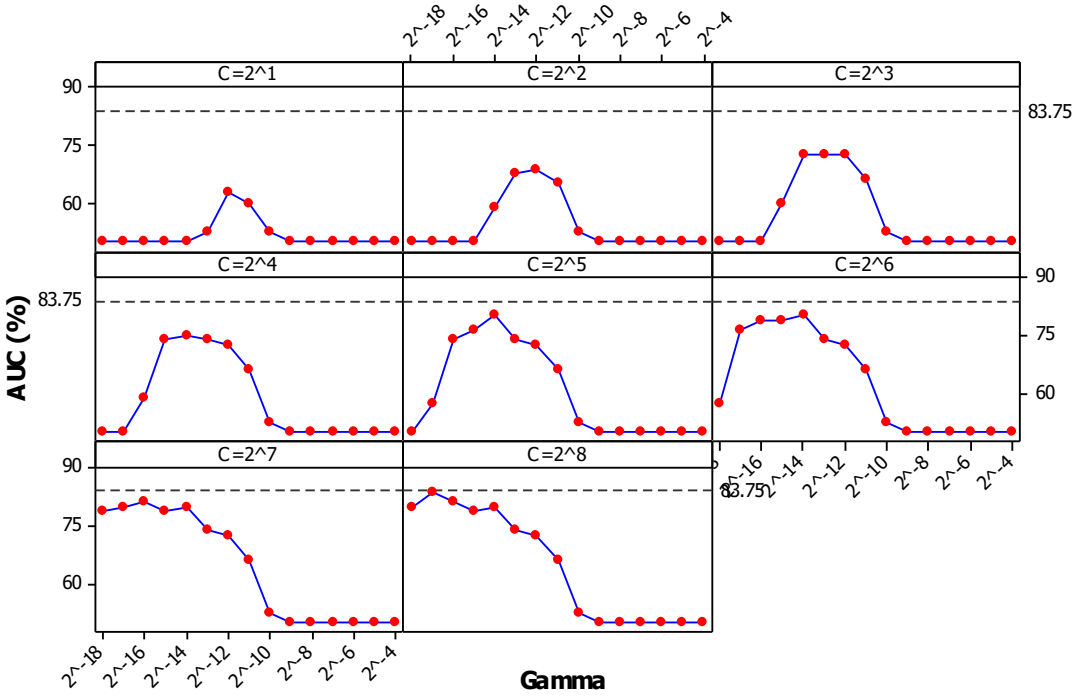
Berdasarkan Gambar 4.14 dapat dilihat bahwa nilai AUC maksimum yang mampu dicapai adalah 100% dan nilai tersebut sudah dicapai saat $C = 2^{9}$. Adapun nilai parameter γ saat $C = 2^{9}$ yang menghasilkan nilai AUC tertinggi adalah $\gamma = 2^{-12}$. Sehingga untuk data *breast cancer* pada *selected feature* dengan menggunakan metode FSVM didapatkan parameter yang optimal yaitu $C = 2^{9}$ dengan $\gamma = 2^{-12}$ dan $C = 2^{10}$ dengan $\gamma = 2^{-13}$. Fungsi pemisah yang terbentuk untuk klasifikasi pada Data *breast cancer* pada *selected feature* menggunakan FSVM adalah

$$f(\mathbf{x}) = \sum_{i=1}^{133}\sum_{j=1}^{133} \alpha_i y_i \exp\left(-2^{-12}\left\|x_i - x_j\right\|^2\right) + b$$

4.6.2 Klasifikasi Data Colon Cancer dengan FSVM

1.) Klasifikasi menggunakan seluruh *feature*

Gambar 4.15 menunjukkan hasil klasifikasi data *colon cancer* dengan seluruh *feature* menggunakan FSVM.



Gambar 4.15 Hasil Klasifikasi *Colon Cancer* Seluruh *Feature* dengan FSVM

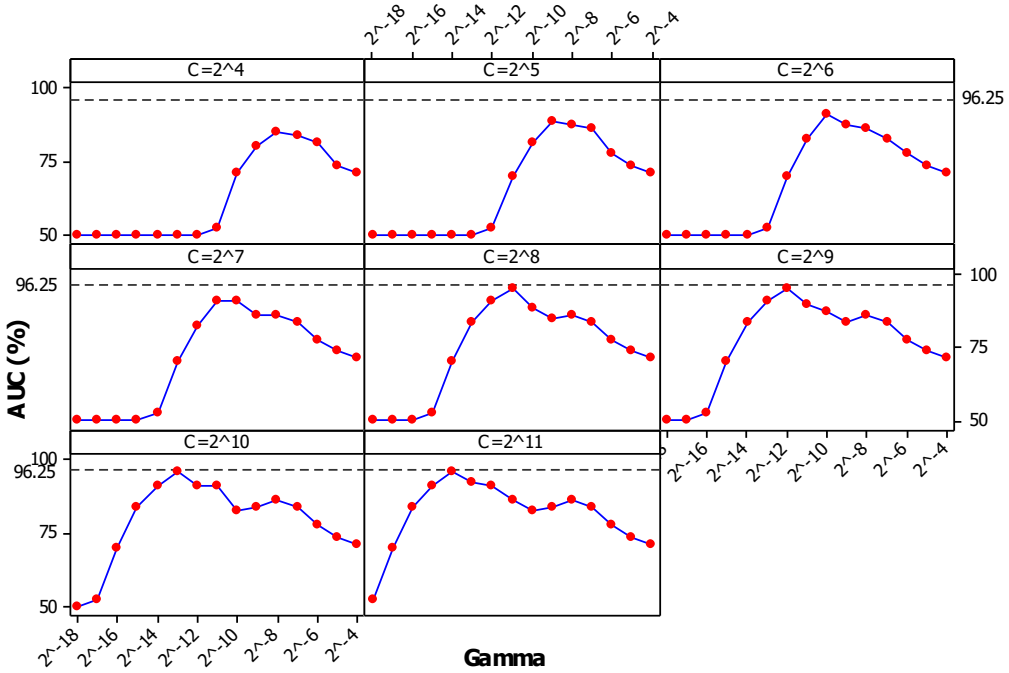
Berdasarkan Gambar 4.15 dapat dilihat bahwa nilai C yang lebih besar menghasilkan AUC yang lebih tinggi. Nilai AUC mencapai maksimum saat C = 2^8 yaitu 83,75%. Adapun nilai parameter γ, saat γ sama dengan 2^{-18} sampai 2^{-13} menghasilkan nilai AUC yang berbeda-beda pada nilai C yang berbeda, namun nilainya mulai mengalami penurunan yang besar saat $\gamma = 2^{-10}$ dan menghasilkan nilai AUC yang kecil sampai $\gamma = 2^{-4}$. Sehingga untuk data colon cancer pada seluruh *feature* dengan menggunakan metode FSVM didapatkan parameter yang optimal yaitu $C = 2^8$ dan $\gamma = 2^{-17}$ dengan nilai rata-rata AUC tertinggi yang mampu dicapai adalah 83,75%.

Fungsi pemisah yang terbentuk untuk klasifikasi pada data *colon cancer* pada seluruh *feature* menggunakan FSVM adalah

$$f(\mathbf{x}) = \sum_{i=1}^{53}\sum_{j=1}^{53} \alpha_i y_i \exp\left(-2^{-17}\left\|x_i - x_j\right\|^2\right) + b$$

2.) Klasifikasi menggunakan *selected feature*

Selain dilakukan klasifikasi menggunakan seluruh *feature* pada data *colon cancer* berikut diberikan hasil klasifikasi data *colon cancer* pada *selected feature* menggunakan FSVM.



Gambar 4.16 Hasil Klasifikasi *Colon Cancer Selected Feature* dengan FSVM

Berdasarkan Gambar 4.16 dapat dilihat bahwa nilai C yang lebih besar menghasilkan AUC yang lebih tinggi. Nilai AUC maksimum yang mampu dicapai adalah 96,25% dan nilai tersebut dicapai saat $C=2^{10}$. Adapun nilai parameter γ untuk $C = 2^{10}$ mencapai nilai maksimum saat $\gamma = 2^{-13}$. Sehingga untuk data colon cancer pada *selected feature* dengan menggunakan metode FSVM didapatkan parameter yang optimal yaitu $C = 2^{10}$ dengan $\gamma = 2^{-13}$ dan $C = 2^{11}$ dengan $\gamma = 2^{-15}$. Fungsi pemisah yang terbentuk untuk klasifikasi pada data *colon cancer* pada *selected feature* menggunakan FSVM adalah

$$f(\mathbf{x}) = \sum_{i=1}^{53}\sum_{j=1}^{53} \alpha_i y_i \exp\left(-2^{-13}\left\|x_i - x_j\right\|^2\right) + b$$

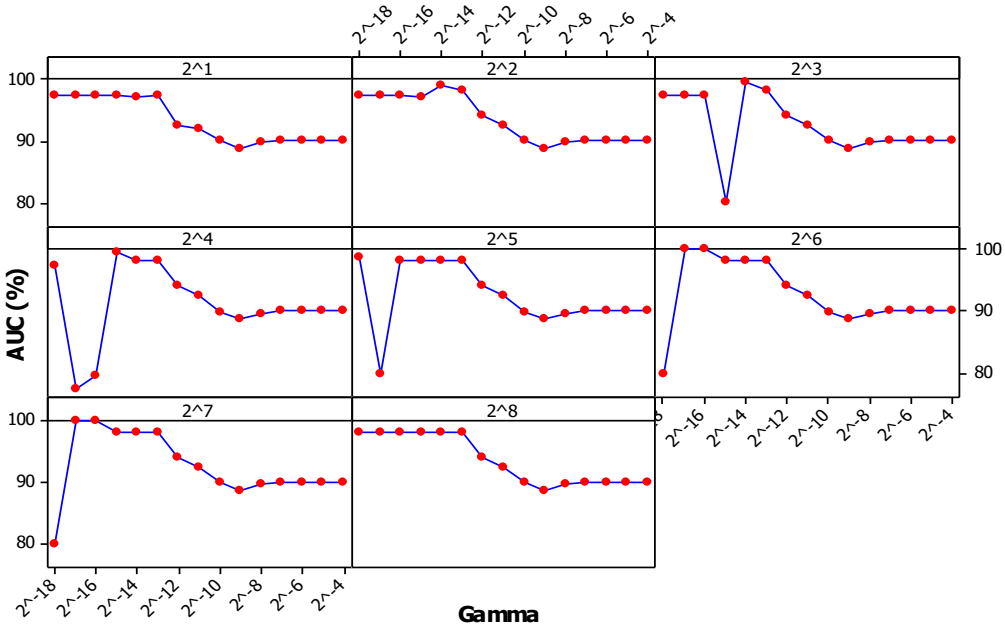
4.7 Klasifikasi dengan EFSVM

Pada klasifikasi dengan EFSVM keanggotaan *entropy* based *fuzzy* pada persamaan (2.37) didapatkan terlebih dahulu. Keanggotaan *entropy* based *fuzzy* tersebut digunakan sebagai input pada klasifikasi EFSVM dimana nilainya dikalikan dengan parameter cost (C). Sama seperti klasifikasi menggunakan SVM dan FSVM, pada EFSVM parameter yang akan dioptimasi adalah parameter cost (*C*) dan gamma (*γ)*. Parameter optimal ditentukan berdasarkan nilai rata-rata AUC tertinggi yang diperoleh dari 5-*fold Cross-validation* dengan kombinasi nilai parameter yang digunakan yaitu *C* pada *range* 2^1, 2^2, 2^3, 2^4, ..., 2^{11} serta nilai parameter γ pada *range* 2^{-18}, 2^{-17}, 2^{-16}, 2^{-15},..., 2^{-4}.

4.7.1 Klasifikasi Data *Breast Cancer* dengan EFSVM

1.) Klasifikasi menggunakan seluruh *feature*

Hasil klasifikasi pada data *breast cancer* dengan seluruh *feature* ditunjukkan pada Gambar 4.17.



Gambar 4.17 Hasil Klasifikasi *Breast Cancer* Seluruh *Feature* dengan EFSVM

Berdasarkan Gambar 4.17 dapat dilihat bahwa nilai AUC hampir sama untuk nilai C yang berbeda-beda. Adapun nilai parameter γ, saat γ sama dengan 2^{-18} sampai 2^{-14} menghasilkan nilai AUC yang tinggi pada C = 2^1, 2^2, dan 2^8. Namun nilainya mulai mengalami penurunan pada semua nilai C saat γ = 2^{-13}. Sehingga untuk data *breast cancer* pada seluruh *feature* dengan menggunakan

metode EFSVM didapatkan parameter yang optimal yaitu C = 2^6 dan C = 2^6 dengan $\gamma = 2^{-15}$ dan $\gamma = 2^{-16}$ dimana nilai rata-rata AUC tertinggi yang mampu dicapai adalah 100%.

Fungsi pemisah yang terbentuk untuk klasifikasi pada data *breast cancer* pada seluruh *feature* menggunakan EFSVM dapat ditulis sebagai berikut.

Diketahui:

$\mathrm{x_i} = [x_{1i}, x_{2i}, x_{3i}, ..., x_{2905i}], i = 1, 2, ..., 134$

Diperoleh nilai *b* dan α_i yang dilampirkan pada Lampiran 17:

b = -0,23364

α_i berukuran 134 × 1 (**α** = 3,243, 0, 3,045, 1,248, 5,85, 0, 4,54, 0, ..., 0)

Maka

$f(\mathbf{x}) = (\mathbf{w}'\mathbf{x} + b)$

dimana $\mathbf{w} = \sum_{i=1}^{134}\sum_{j=1}^{134} \alpha_i y_i K(x_i, x_j)$ dan

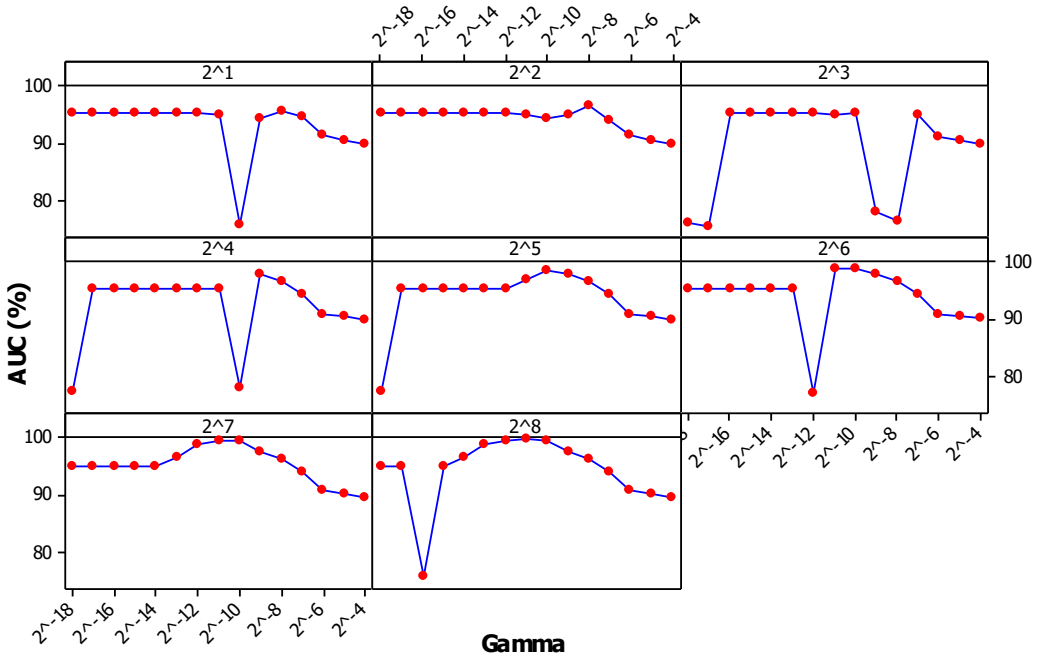
$$K(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) = \exp\left(-2^{-15} \left\| x_i - x_j \right\|^2\right)$$

Sehingga persamaan untuk fungsi pemisah adalah:

$$f(\mathbf{x}) = \sum_{i=1}^{134}\sum_{j=1}^{134} \alpha_i y_i \exp\left(-2^{-15} \left\| x_i - x_j \right\|^2\right) - 0,23364$$

2.) Klasifikasi menggunakan *selected feature*

Selain dilakukan klasifikasi menggunakan seluruh *feature* pada data *breast cancer* berikut diberikan hasil klasifikasi data *breast cancer* pada *selected feature* menggunakan EFSVM. Berdasarkan Gambar 4.18 dapat dilihat bahwa nilai AUC maksimum yang mampu dicapai adalah 100% dan nilai tersebut sudah dicapai saat C = 2^8. Adapun nilai parameter γ saat C = 2^8 yang menghasilkan nilai AUC tertinggi adalah $\gamma = 2^{-11}$. Sehingga untuk data *breast cancer* pada *selected feature* dengan menggunakan metode EFSVM didapatkan parameter yang optimal yaitu C = 2^8 dan $\gamma = 2^{-11}$.



Gambar 4.18 Hasil Klasifikasi *Breast Cancer Selected Feature* dengan EFSVM

Fungsi pemisah yang terbentuk untuk klasifikasi pada data *breast cancer* pada *selected feature* menggunakan EFSVM sebagai berikut.

Diketahui:

$\mathrm{x}_i = [x_{1i}, x_{2i}, x_{3i}, ..., x_{51i}], i = 1, 2, ..., 134$

Diperoleh nilai b dan α_i yang dilampirkan pada Lampiran 17 :

b = -0,21368

α_i berukuran 134 × 1 ($\boldsymbol{\alpha}$ = 0,99, 0,29, 0,151, 0,815, 0,724, 0,643, 0,17,..., 0,75)

Maka

$f(\mathbf{x}) = (\mathbf{w}'\mathbf{x} + b)$

dimana $\mathbf{w} = \sum_{i=1}^{134}\sum_{j=1}^{134} \alpha_i y_i K(x_i, x_j)$ dan

$$K(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) = \exp\left(-2^{-11} \left\| x_i - x_j \right\|^2\right)$$

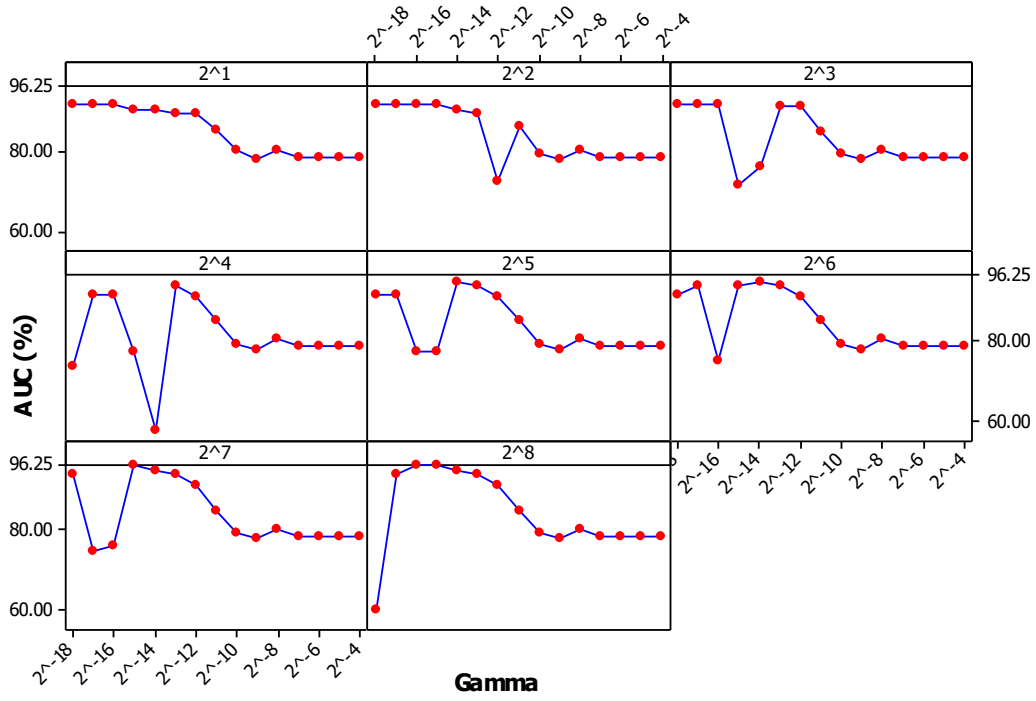
Sehingga persamaan untuk fungsi pemisah adalah:

$$f(\mathbf{x}) = \sum_{i=1}^{134}\sum_{j=1}^{134} \alpha_i y_i \exp\left(-2^{-11} \left\| x_i - x_j \right\|^2\right) - 0,21368$$

4.7.2 Klasifikasi Data Colon Cancer dengan EFSVM

1.) Klasifikasi menggunakan seluruh *feature*

Hasil klasifikasi pada data *colon cancer* dengan seluruh *feature* ditunjukkan pada Gambar 4.19.



Gambar 4.19 Hasil Klasifikasi *Colon Cancer* Seluruh *Feature* dengan EFSVM

Berdasarkan Gambar 4.19 dapat dilihat bahwa nilai AUC maksimum yang mampu dicapai adalah 96,25% dan nilai tersebut dicapai saat C = 2^7. Adapun nilai parameter γ saat C = 2^6 yang menghasilkan nilai AUC tertinggi adalah γ = 2^{-15}. Sehingga untuk data colon cancer pada seluruh *feature* dengan menggunakan metode EFSVM didapatkan parameter yang optimal yaitu C = 2^7 dan γ = 2^{-15}. Fungsi pemisah yang terbentuk untuk klasifikasi pada data *colon cancer* pada seluruh *feature* menggunakan EFSVM sebagai berikut.

Diketahui:

$$x_i = [x_{1i}, x_{2i}, x_{3i}, ..., x_{2900i}], i = 1, 2, ..., 50$$

Diperoleh nilai *b* dan α_i yang dilampirkan pada Lampiran 18:

b = 0,09

α_i berukuran 50 × 1 (**α** = 18,84, 4,92, 69,7, 44,4, 3,77, 8,88, 0,..., 19,2)

Maka

$$f(\mathbf{x}) = (\mathbf{w}'\mathbf{x} + b)$$

dimana $\mathbf{w} = \sum_{i=1}^{50}\sum_{j=1}^{50} \alpha_i y_i K(x_i, x_j)$ dan

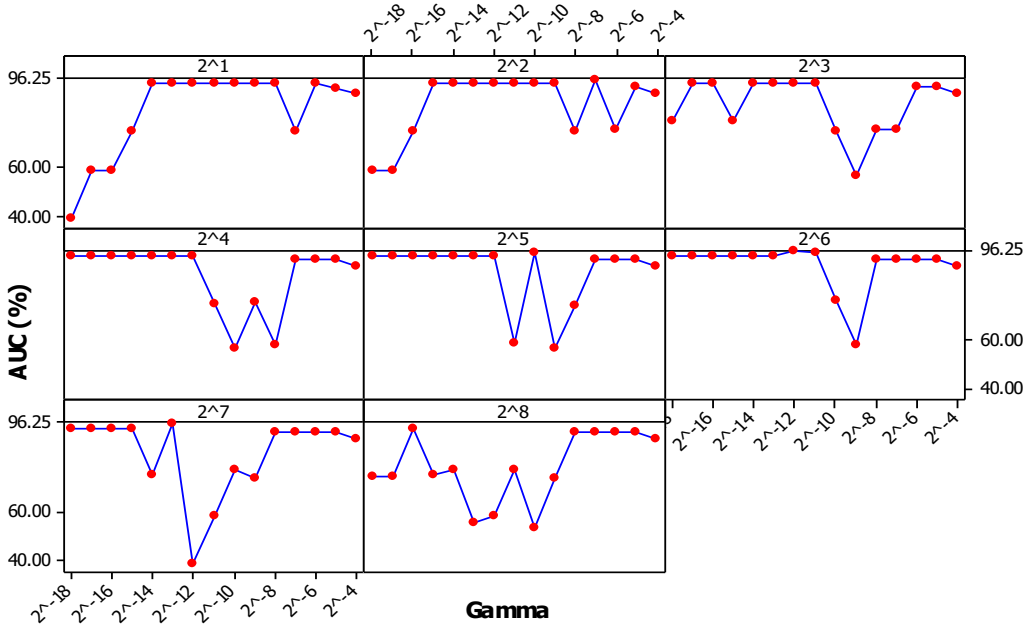
$$K(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) = \exp\left(-2^{-15} \left\| x_i - x_j \right\|^2\right)$$

Sehingga persamaan untuk fungsi pemisah adalah:

$$f(\mathbf{x}) = \sum_{i=1}^{50}\sum_{j=1}^{50} \alpha_i y_i \exp\left(-2^{-15} \left\| x_i - x_j \right\|^2\right) + 0,09$$

2.) Klasifikasi menggunakan *selected feature*

Selain dilakukan klasifikasi menggunakan seluruh *feature* pada data colon cancer berikut diberikan hasil klasifikasi data colon cancer pada *selected feature* menggunakan EFSVM.



Gambar 4.20 Hasil Klasifikasi *Colon Cancer* Selected *Feature* dengan EFSVM

Gambar 4.20 menunjukkan hasil klasifikasi pada data colon cancer selected feature. Berdasarkan Gambar 4.20 dapat dilihat bahwa nilai AUC maksimum yang mampu dicapai adalah 96,25% dan nilai tersebut dicapai saat C = 2^7 dan C=2^6 . Adapun nilai parameter γ saat C = 2^7 dan C = 2^6 yang menghasilkan nilai AUC tertinggi adalah γ = 2^{-13}. Sehingga untuk data colon cancer pada *selected*

feature dengan menggunakan metode EFSVM didapatkan parameter yang optimal yaitu C = 2^7 dengan $\gamma = 2^{-13}$ dan C = 2^6 dengan $\gamma = 2^{--13}$.

Fungsi pemisah yang terbentuk untuk klasifikasi pada data *colon cancer* pada *selected feature* menggunakan EFSVM sebagai berikut.

Diketahui:

$\mathrm{x_i} = [x_{1i}, x_{2i}, x_{3i}, ..., x_{15i}], i = 1, 2, ..., 50$

Diperoleh nilai *b* dan α_i yang dilampirkan pada Lampiran 18:

b = -0,05882

α_i berukuran 50 × 1 (**α** = 1,6, 1,6 2, 2, 2, 1,6, 0, 0, 1, 0,..., 2)

Maka

$f(\mathbf{x}) = (\mathbf{w}'\mathbf{x} + b)$

dimana $\mathbf{w} = \sum_{i=1}^{50}\sum_{j=1}^{50} \alpha_i y_i K(x_i, x_j)$ dan

$$K(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) = \exp\left(-2^{-13}\left(x_i - x_j\right)^2\right)$$

Sehingga persamaan untuk fungsi pemisah adalah:

$$f(\mathbf{x}) = \sum_{i=1}^{50}\sum_{j=1}^{50} \alpha_i y_i \exp\left(-2^{-13}\left(x_i - x_j\right)^2\right) - 0,05882$$

Setelah didapatkan parameter terbaik dengan nilai AUC (%) tertinggi pada masing-masing data, berikut diberikan rangkuman hasil performansi klasifikasi EFSVM.

Tabel 4.15 Rangkuman Performansi Klasifikasi EFSVM

Datasets	*Features*	AUC	Akurasi	*Sensitivity*	*Specificity*	*G-means*
Breast	Semua *features*	100	100	100	100	100
Cancer	*Selected features*	100	100	100	100	100
Colon	Semua *features*	96.25	94	100	92.5	95.81
Cancer	*Selected features*	96.25	95.38	100	92.5	96.02

Berdasarkan Tabel dapat dilihat bahwa klasifikasi pada *breast cancer* menghasilkan nilai 100% pada seluruh perfomansi klasifikasi. Hal tersebut berarti seluruh sampel pada data *breast cancer* telah diklasifikasikan dengan benar. Fungsi pemisah yang telah terbentuk mampu membedakan apakah sampel

tersebut masuk dalam kategori "*good*" atau "*poor*". Sementara itu untuk data *colon cancer* didapatkan akurasi 94% yang berarti kemampuan fungsi pemisah dalam membedakan sampel berdasarkan kelasnya adalah 94% dan sisanya berarti mausk dalam kesalahan klasifikasi. Nilai *sensitivity* sama dengan 100% menunjukkan fungsi pemisah mampu mengidentifikasi sampel yang berasal dari kelas "tumor" dengan benar. *Specificity* sama dengan 92,5% berarti kemampuan fungsi pemisah untuk mengidentifikasi sampel dari kelas "normal" adalah 92,5% dan sisanya (7,5%) merupakan kesalahan klasifikasi, terdapat sampel dari kelas "normal" namun dikelompokkan menjadi kelas "tumor". AUC adalah fungsi *sensitivity* dan *spesificity* yang dapat merangkum nilai-nilai tersebut bersama. Pada data *colon cancer*, AUC sebesar 96,25% menunjukkan kemampuan fungsi pemisah dalam membedakan kelas "tumor" atau "normal" sudah baik. Selain itu penggunaan semua *feature* dan *selected feature* pada *breast cancer* keduanya menghasilkan performansi maksimal 100%. Sedangkan pada *colon cancer*, meski berdasarkan akurasi dan *G-means* didapatkan hasil lebih tinggi untuk *selected feature* namun secara keseluruhan keduanya memiliki performa yang sama karena nilai AUC sama yaitu 96,25%.

4.8 Perbandingan Performansi Klasifikasi

Setelah dilakukan analisis pada data *breast cancer* dan colon cancer menggunakan SVM, FSVM, dan EFSVM baik untuk data dengan seluruh *feature* maupun *selected feature*, pada bagian ini akan dilakukan perbandingan dari masing-masing metode. Perbandingan beberapa metode tersebut dinyatakan dalam performansi klasifikasi yang meliputi AUC, Akurasi, *Sensitivity*, *Specificity* dan *G-means* yang merupakan hasil dari klasifikasi dengan parameter terbaik dari masing-masing metode. Perbandingan hasil klasifikasi tersebut terdapat pada Tabel 4.15.

Berdasarkan Tabel 4.15, klasifikasi *breast cancer* menggunakan SVM dengan seluruh *feature* didapatkan nilai *sensitivity* 100% namun *specificity* 0%. Nilai tersebut berarti klasifikasi dengan SVM mampu mengklasifikasikan 22 data *testing* untuk kelas negatif (mayoritas) dengan benar namun tidak bisa mengklasifikasi 12 data *testing* untuk kelas positif (minoritas) dengan benar. Sebanyak 12 data *testing* yang seharusnya masuk pada kelas positif justru

diklasifikasikan masuk pada kelas negatif. SVM belum baik dalam mengidentifikasi sampel dengan kategori "poor" sehingga terdapat kemungkinan yang tinggi bahwa sampel dalam keadaan "poor" namun dikategorikan pada kelas "good". Adanya kasus *imbalanced* pada data *breast cancer* ini menjadi penyebab rendahnya nilai *specificity*. Selain itu untuk *G-means* didapatkan nilai 0% yang disebabkan karena performansi yang rendah dari kelas positif (minoritas), jadi walaupun 22 data testing pada kelas negatif diklasifikasikan dengan benar namun 12 data testing pada kelas positif diklasifikasikan salah maka nilai *G-means* akan rendah. Setelah diterapkan FSVM dan EFSVM pada data *breast cancer* dengan menggunakan semua *feature*, EFSVM mampu menghasilkan performansi klasifikasi terbaik dibandingkan dengan SVM dan FSVM yaitu 100%, artinya semua data testing diklasifikasikan dengan benar. Selain itu untuk data *breast cancer* dengan *selected feature*, FSVM dan EFSVM menghasilkan nilai performansi yang sama yaitu 100%. Penggunaan hanya *selected feature* juga terbukti mampu meningkatkan nilai performansi klasifikasi pada metode SVM dan FSVM, sedangkan pada EFSVM baik seluruh *feature* maupun *selected feature* dihasilkan performansi sempurna yaitu 100%.

Tabel 4.16 Perbandingan Hasil Performansi Klasifikasi

Data	*feature*	Metode	AUC (%)	Akurasi	*Sensitivity*	*Specificity*	*G-means*
Breast Cancer	Semua *feature*	SVM	73.36	66.14	100	0	0
		FSVM	99.56	99.42	100	99.13	99.56
		EFSVM	**100**	**100**	**100**	**100**	**100**
	Selected feature	SVM	82.54	83	83.98	81.11	81.77
		FSVM	**100**	**100**	**100**	**100**	**100**
		EFSVM	**100**	**100**	**100**	**100**	**100**
Colon Cancer	Semua *feature*	SVM	86.25	87.69	92.5	80	85.18
		FSVM	83.75	79.08	100	67.5	79.567
		EFSVM	**96.25**	**94**	**100**	**92.5**	**95.811**
	Selected feature	SVM	88.25	89.23	92.5	84	87.58
		FSVM	**96.25**	**95.38**	**100**	**92.5**	**96.028**
		EFSVM	**96.25**	**95.38**	**100**	**92.5**	**96.028**

Sementara itu pada data *Colon Cancer* EFSVM memberikan nilai performansi klasifikasi yang paling tinggi, baik pada data dengan menggunakan seluruh *feature* mapun *selected feature*. Pada data dengan *selected feature* dihasilkan nilai performansi klasifikasi yang sama tinggi antara metode FSVM

dan EFSVM. Sama seperti pada data *breast cancer*, adanya *feature selection* terbukti mampu meningkatkan nilai performansi klasifikasi pada data *colon cancer.*

BAB 5
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisis yang telah dilakukan dapat disimpulkan sebagai berikut.

1. Dalam melakukan klasifikasi *imbalanced* data menggunakan EFSVM terdapat dua tahapan utama yang dilakukan yaitu menentukan *entropy based fuzzy membership* dan mendapatkan performansi klasifikasi EFSVM pada tiap kombinasi nilai parameter. Parameter terbaik dari klasifikasi EFSVM dipilih berdasarkan nilai performansi yang paling tinggi. Pada data *breast cancer* didapatkan paramter optimal adalah C = 2^2 dan $\gamma = 2^{-14}$ dengan nilai rata-rata AUC 100% dan untuk data *colon cancer* parameter optimal adalah C = 2^6 dan $\gamma = 2^{-15}$ dengan nilai AUC 96,25%.
2. Hasil perbandingan metode klasifikasi dengan menggunakan SVM, FSVM, dan EFSVM memberikan hasil bahwa metode EFSVM lebih unggul dibandingkan dengan SVM dan FSVM untuk kedua dataset. Namun setelah diterapkan *feature selection*, didapatkan hasil performansi klasifikasi yang hampir sama antara FSVM dan EFSVM pada kedua data.

5.2 Saran

Berdasarkan hasil analisis serta kesimpulan yang diperoleh terdapat beberapa hal yang disarankan untuk penelitian selanjutnya adalah

1. Pada penelitian selanjutnya, diharapkan menggunakan data dengan nilai *imbalanced ratio* yang berbeda-beda sehingga dapat diketahui bagaimana penerapan EFSVM pada data dengan nilai IR tertentu
2. Perlu dilakukan perbandingan menggunakan metode lainnya dalam menangani kasus data imbalanced

(Halaman ini sengaja dikosongkan)

DAFTAR PUSTAKA

Akbani, R., Kwek, S., & Japkowicz, N. (2004). Applying Support Vector Machines to Imbalanced Datasets . *European Conference on Machine Learning, Springer*, 39-50.

Alonso, A., Noelia, S., & Veronica, B. (2015). Feature Selection for High Dimensional Data. *Artificial Intelligence: Fondations, Theory, and Algorthms. Springel International Publishing Switzerland*.

Batista, G., C., P., & C, M. (2004). A Study of the Behavior of Several Methods fo Balancing Machine Learning Training Data. *ACM SIGKDD Explorations Newsletter*, 20-29.

Batista, G., Ronaldo, C., & Monard, M. (2004). A Study of the Behavior of Several Methods fo Balancing Machine Learning Training Data. *ACM SIGKDD Explorations Newsletter*, 20-29.

Batuwita, R., & Palade, V. (2010). FSVM-CIL: Fuzzy Support Vector Machines fo Class Imbalance Learning. *IEEE Trans. Fuzzy Syst.* , 558-571.

Bekkar, M., Djema, H. K., & Alitouche, T. A. (2013). Evaluation Measures for Models Assesment over Imbalanced Datasets. *Journal of Information Engineering and Application*, 27-38.

Berrar, D. P., Dubitzky, W., & Granzow, M. (2003). *A Practical Approach to Microarray Data Analysis.* Dordrecht: Kluwer Academic Publisher.

Bowyer, K., Chawla, N., Hall, L., & Kegelmeyer, W. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal Artificial Intelligent*, 321-357.

Brieman, L., Friedman, J., Olshen, R., & C, S. (1984). *Classification and Regression Trees.* Wadsworth International Group.

Brown, I., & Mues, C. (2012). An Experimental Comparison of Classification Algorthms for Imbalanced Credit Scoring Datasets. *Expert System Application*, 3446-3453.

Brown, M. P., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C. W., Furey, T. S., et al. (2000). Knowledge-based analysis of microarray gene expression data by using support vector machines. *PNAS*, 262-267.

Byvatov, E. (2003). Comparison of Support Vector Machine and Artificial Neural Network System for Drug/Nondrug classification. *Chem Inf Compu Sci*, 1882-1889.

Canedo, V. B., Marono, N. S., Betanzos, A. A., Benitez, J., & Herrera, F. (2014). A Review of Microarray Datasets and Applied Feature Selection Methods. *information Science*, 111-135.

Chauduri, A., & De, K. (2011). Fuzzy Support Vector Machine for Bankcruptcy Prediction. *Appl. Soft Computing*, 2472-2486.

Chawla, N., Cieslak, D., Hall, L., & Joshi, A. (2008). Automatically Countering Imbalanced and its Empirical Relationship to Cost. *Data Mining Knowledge Discovery*, 225-252.

Choi, J. (2010). A Selective Sampling Method for Imbalanced Data Learning on Support Vector Machines. *Graduate Theses and Dissertations, Paper 11529*.

Chu, F., & Lipo, W. (2005). Applications of Support Vector Machines to Cancer Classification with Microarray Data. *International Journal of Neiral System*, 475-484.

Cortes, C., & Vapnik, V. (1995). Support-Vector Network. *Machine Learning, 20,*, 273-297.

Dai, H. (2015). Class Imbalanced Learning via a Fuzzy Total Margin based Support Vector Machine. *Applied Soft Computing*, 31:172-184.

Deng, X., & Tian, X. (2013). Non Linear Process Pattern Recognition Using Statistics Kernel PCA Similiarity Factor. *Neurocomputing*, 121:298-308.

Dudoit, S., Fridlaynd, J., & Speed, T. P. (2014). Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data. *Journal of the American Association*, 77-87.

Fan, Q., Wang, Z., Li, D., Gao, D., & Zha, H. (2016). Entropy-based Fuzzy Support Vector Machine for Imbalanced Datasets. *Knowledge-Based System*, 87-99.

Galar, M., Fernandez, A., Barrenchea, E., Bustince, H., & Herrera, F. (2012). A review on Ensembles for the Class Imbalanced problem: bagging-, boosting-, and Hybrid-based Approach. *IEEE Trans. Syst. Man Cybern*, 465-484.

Galar, M., Fernandez, A., Barrenechea, H., & Herrera, F. (2013). Eusboost: Enchancing Ensembles for Highliy Imbalanced Datasets by Evolutionary Undersampling. *Pattern Recognit*, 3460-3471.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., P, M. J., et al. (1999). Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science 286*, 531-537.

Gunn, S. (1998). *Support vector Machines fo Classification and Regression.* Technical Report, ISIS.

Guo, J., Yi, P., Wang, R., Ye, Q., & Zhao, C. (2014). Feature Selection for Least Square Projection Twin Support Vector Machine . *Neurocomputing*, Vol. 14, Hal. 174-183.

Guo, Y., & Zhang, H. (2014). Oil Spill Detection Using Synthetic Aperture Radar Images and Feature Selection in Shape Space. *International Journal of Applied Earth Observation and Geoinformation*, 30:146-157.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning*, 389-422.

Han, J., & Kamber, M. (2001). *Data Mining: Concepts and Techniques* . USA: Academic Press.

Han, J., Kamber, M., & Jian, P. (2006). *Data Mining: Concept and Techniques (3th ed).* San Fransisco: Morgan Kaufmaan.

Hand, D., & Till, R. (2001). A Simple generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. *Machine Learning, vol. 45*, 171-186.

Hardle, W. K., & Simar, L. (2015). *Applied Multivariate Statistical Analysis.* Verlag Berlin Heidelberg: Springer.

Hong, J.-H., & Cho, S.-B. (2006). The Classification of Cancer based on DNA Microarray Data that Uses Diverse Ensemble Genetic Programming. *Artificial Intelligence in Medicine*, 43-58.

Hsu, C., Chang, C., & Lin, C. (2004). *A Practical Guide to Support Vector Classification.* Department of Computer Science and Information Engineering, National Taiwan University.

Huang, C. L., & Huang, S. (2007). Model Selection for Support Vector machine via uniform Design. *Computational Statistic and Data Analysis*, vol 52, 335-346.

Khaulasari, H. (2016). *Combine Sampling-Least Square Support Vector Machine untuk Klasifikasi Multi Class Imbalanced Data.* Program Pascasarjana, Institut Teknologi Sepuluh Nopember, Surabaya.

Kubat, M., & Matwin, S. (1997). Addressing the Curse of Imbalanced Training Sets: One-sided Selection. *International Conference on Machine Learning*, 179-186.

Lee, J. W., Lee, J. B., Park, M., & Song, S. H. (2005). An extensive comparison of recent classification tools applied to microarray data. *Computational Statistics & Data Analysis*, 869-885.

Lee, M., & To, C. (2010). Comparison of Support Vector Machine and Back Propagation Neural Network in Evaluating the Enterprise Financial Distress. *International Journal of Artificial & Application (IJAIA), Vol. 1, No. 3, (July, 2010)*.

Liao, J., & Chin, K.-V. (Vol. 23 no. 15 2007). Logistic Regression for Disease Classification Using Microarray Data: Model Selection in a Large p and Small n Case. *Bioinformatics*, 1945-1951.

Lin, C., & Wang, S. (2002). Fuzzy Support Vector Machines. *IEEE Trans. Neural Network* , 464-471.

Liu, H., & Lei, Y. (2003). Feature Selection for High Dimensional Data : A Fast Correlation Based Filter Solution. *Proceeding of Twentieth International Conference on Machine Learning (ICML-2003)*.

Liu, X., Wu, J., & Zhou, Z. (2009). Exploratory Undersampling for Class-imbalanced Learning. *IEEE Trans. Syst. Man Cybern*.

Maldonado, S., & Lopez, J. (2014). Imbalanced data Classification using Second-order Cone Programming Support Vector Machines. *Pattern Recognit*, 2070-2079.

Mercer, J. (1909). Fondation of Positive and Negative Type and Their Connection with the Theory of Integral Equations. *Philosophical Transactions of the Royal Society of London*, Vol. 25, Hal. 3-23.

Newton, I. (1729). *In Experimental Philosophy Particular Propositions are Inferred from the Phenomena and Afterwards Rendered General by Induction.* 3rd ed.: Adrew Motte's English Translation Published, Vol.2.

Ozcift, A., & Gulten, A. (2011). Classifier Ensemble Construction with Rotation Forest to Improve Medical Diagnosis Performance of Machine Learning Algorthms. *Computer Methods and Programs in Biomedicine*, 104(3):443-451.

Rahman, A., Smith, D., & TImms, G. (2014). A Novel Machine Learning Approach Toward Quality Assesment of Sensor Data. *IEEE Sensors Journal*, 14(4):1035-1047.

Rahman, F., & Purnami, S. (2012). Perbandingan Klasifikasi Tingkat Keganasan Breast Cancer dengan Menggunakan Regresi Logistik Ordinal dan Support Vector Machine. *Jurnal SAINS dan Seni ITS, Vol. 1, No.1, (September 2012) ISSN : 2301-928X*.

Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Chen-Hsiang, Y., Angelo, M., et al. (2001). Multiclass cancer diagnosis using tumor gene expression signatures. *PNAS*, 15149-15154.

Ramon, D.-U., & Sara, A. d. (2006). Gene Selection and Classification of Microarray Data using Random Forest. *BMC Bioinformatics*.

Saeys, J., Inza, I., & Larranaga, P. (2007). A Review of Feature Selection Techniques in Bioinformatics. *Bioinformatics 23 (19)*, 2507-2517.

Santosa, B. (2007). *Data Mining: teknik Pemanfaatan Data untuk Keperluan Bisnis, teori da Aplikasi.* Graha Ilmu.

Scholkopf, B., & Smola, A. (2002). *Learning with Kernel: Support Vector Machines, Regulerization, Optimization, and Beyond.* Cambridge: MA: MIT Press.

Seeja, K. R., & Shweta. (2011). Microarray Data Classification Using Support Vector Machine. *International Journal of Biometrics and Bioinformatics (IJBB)*, Volume (5) : Issue (1) : 10-15.

Shannon, C. (2001). A Mathematical Theory of Communication . *SIGMOBILE Mobil Comput. Commun.Rev. 5 (1)*, 3-55.

Shapiro, G., & Tamayo, P. (2003). Microarray Data Mining: Facing the Chalenges. *ACM SIGKDD Explo. Newsl. 5(2)*, 1-5.

Sun, Y., Wong, A., & Kamel, M. (2009). Classification of Imbalanced Data : A Review. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(4):687-719.

Tan, P., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining (4th ed.)*. Boston: Pearson Addison Wesley.

Tian, D., Peng, G., & Ha, M. (2012). Fuzzy Support Vector Machine Based on Non-equilibrium data. *International Conference on Machine Learning and Cybernetics, 2, IEEE*, 448-453.

Trapsilasiwi, R. K. (2013). *Klasifikasi Multiclass untuk Imbalanced Data Menggunakan SMOTE Least Square Support Vector Machine*. Program Pascasarjana, Institut Teknologi Sepuluh Nopember, Surabaya.

Wang, S., Li, D., Song, X., Wei, Y., & Li, H. (2011). A Feature Selection Method Based On Improved Fisher's Discriminant Ratio for Text Sentiment Classification . *Expert System with Applicatios*, 8696-8702.

Wang, Y., Wang, S., & Lai, K. (2005). A New Fuzzy Vector Machine to Evaluate Credit Risk. *IEEE Trans. on Fuzzy Syst.*, 13(6):820-831.

Yohannes, Y., & Webb, P. (1999). Classification and Regression Trees, A user manual for identifying of Vulnerability to famine and chronic food insecurity. *Microcompoter in Policy Research International Food Policy Research Institute, Washington DC, USA*.

(Halaman ini sengaja dikosongkan)

LAMPIRAN

Lampiran 1. Syntax Klasifikasi dengan SVM

```
library(kernlab)
library(e1071)
Kfold = function(y, k) {
  y = as.vector(as.matrix(y))
  datax = data.frame(y = y, idx = c(1:length(y)))
  n = length(y)
  nk = ceiling(n/k)
  datak = vector("list", k)
  levely = as.numeric(levels(as.factor(y)))
  for (j in 1:length(levely)) {
    datai  = datax[which(datax[,1]==levely[j]), ]
    nc     = dim(datai)[1]
    nck    = ceiling(nc/k)

    # acak data setiap setiap kelas
    set.seed(2222)
    sam       = sample(nc, replace = F)
    sam_datai = datai[sam,]
    for (i in 1:k) {
      if (i==k) {
        datak[[i]]        =        c(datak[[i]],sam_datai[(((i-
1)*nck)+1):nc,2])
      } else {
        datak[[i]]        =        c(datak[[i]],sam_datai[(((i-
1)*nck)+1):(i*nck),2])
      }
    }
  }
  return(datak)
}

# grid search
gridSearchx = function(y, x, C, Sigma, fold_sample) {

  #------------------#
  #    ACCURACY
  #------------------#
  pred<-function(x,lable){
    C=NULL
    n = length(x)
    for (i in 1:n){
      if(x[i]>0){C[i]=1}
      else {C[i]=-1}
    }

    TP = 0
    for (i in 1:n) {if (C[i]==1 & lable[i]==1) (TP=TP+1)}

    FN = 0
    for (i in 1:n) {if (C[i]!=1 & lable[i]==1) (FN=FN+1)}

    FP = 0
    for (i in 1:n) {if (C[i]==1 & lable[i]!=1) (FP=FP+1)}
```

```
  TN = 0
  for (i in 1:n) {if (C[i]!=1 & lable[i]!=1) (TN=TN+1)}

  Sensi = TP/(TP+FN)
  Spesi = TN/(TN+FP)
  Gmean = sqrt(Sensi*Spesi)
  AUC   = (1+(TP/(TP+FN))-(FP/(FP+TN)))/2

  accuracy=mean(C==lable)
  result=list(C,accuracy)
  conmat=table(C,lable) #confusion matrix
  list(accuracy=accuracy, conmat=conmat,
       Sensi = Sensi, Spesi = Spesi,
       Gmean = Gmean, AUC = AUC)
}

y           = as.numeric(as.matrix(y))
x           = x
C           = C
Sigma       = Sigma
fold_sample = fold_sample
nf          = length(fold_sample)
nC          = length(C)
nS          = length(Sigma)

result = vector("list", nC)

for (i in 1:nC){
  # Matrix Akurasi
  mat.has = matrix(0, nrow = nf, ncol = nS)
  colnames(mat.has) = Sigma
  rownames(mat.has) = c(1:nf)

  # Matrix Spesi
  Spesi = matrix(0, nrow = nf, ncol = nS)
  colnames(Spesi) = Sigma
  rownames(Spesi) = c(1:nf)

  # Matrix Sensi
  Sensi = matrix(0, nrow = nf, ncol = nS)
  colnames(Sensi) = Sigma
  rownames(Sensi) = c(1:nf)

  # Matrix Gmean
  Gmean = matrix(0, nrow = nf, ncol = nS)
  colnames(Gmean) = Sigma
  rownames(Gmean) = c(1:nf)

  # Matrix AUC
  AUCx = matrix(0, nrow = nf, ncol = nS)
  colnames(AUCx) = Sigma
  rownames(AUCx) = c(1:nf)

  # Matrix CV
  CV = matrix(0, nrow = 1, ncol = nC)
  colnames(CV) = Sigma
  #n = 0
  conmat = vector("list", nS)
```

```
    for (k in 1:nS) {
      CVx = 0
      Mconmat = matrix(0,nrow = nf, ncol = 4)
      colnames(Mconmat) = c("TN", "FN", "FP", "TP")
      #n = n+1
      for (j in 1:nf) {
        xj = x[-fold_sample[[j]],]
        yj = y[-fold_sample[[j]]]

        hasil =try(svm(xj,yj,cost=C[i],gamma=Sigma[k]))

          Xj = x[fold_sample[[j]],]
          Yj = y[fold_sample[[j]]]
          fx=predict(hasil,Xj)
          akurasi=pred(fx,Yj)
          conmatx  = akurasi$conmat
          for (con in 1:length(conmatx)) {
            Mconmat[j,con] = conmatx[con]
          }
          mat.has[j,k] = akurasi$accuracy
          Spesi[j,k]   = akurasi$Spesi
          Sensi[j,k]   = akurasi$Sensi
          Gmean[j,k]   = akurasi$Gmean
          AUCx[j,k]    = akurasi$AUC

        }
      }
    result[[i]] = list(Akurasi = mat.has,
                       Sensitivity = Sensi,
                       Specificity = Spesi,
                       Gmeans = Gmean,
                       AUC = AUCx)
  }
  names(result) = C
  return(result)
}
```

Lampiran 2. Syntax Klasifikasi dengan FSVM

```
# grid search
gridSearchx = function(y, x, C, Sigma, fold_sample) {
  #------------------#
  #      FUZZY
  #------------------#
  Si = function(y) {
    n = length(y)
    s = NULL
    for (i in 1:n){
      if (y[i] == 1) {
        s[i] = 1
      } else {
        s[i] = 0.1
      }
    }
    return(s)
  }
  #------------------#
  #    ACCURACY
  #------------------#
  pred<-function(x,lable){
    C=NULL
    n = length(x)
    for (i in 1:n){
      if(x[i]>0){C[i]=1}
      else {C[i]=-1}
    }

    TP = 0
    for (i in 1:n) {if (C[i]==1 & lable[i]==1) (TP=TP+1)}

    FN = 0
    for (i in 1:n) {if (C[i]!=1 & lable[i]==1) (FN=FN+1)}

    FP = 0
    for (i in 1:n) {if (C[i]==1 & lable[i]!=1) (FP=FP+1)}

    TN = 0
    for (i in 1:n) {if (C[i]!=1 & lable[i]!=1) (TN=TN+1)}

    Sensi = TP/(TP+FN)
    Spesi = TN/(TN+FP)
    Gmean = sqrt(Sensi*Spesi)
    AUC   = (1+(TP/(TP+FN))-(FP/(FP+TN)))/2

    accuracy=mean(C==lable)
    result=list(C,accuracy)
    conmat=table(C,lable) #confusion matrix
    list(accuracy=accuracy, conmat=conmat,
         Sensi = Sensi, Spesi = Spesi,
         Gmean = Gmean, AUC = AUC)
  }

  y           = as.numeric(as.matrix(y))
  x           = x
  C           = C
```

```
  Sigma       = Sigma
  fold_sample = fold_sample
  nf          = length(fold_sample)
  nC          = length(C)
  nS          = length(Sigma)


  result = vector("list", nC)

  for (i in 1:nC){
    # Matrix Akurasi
    mat.has = matrix(0, nrow = nf, ncol = nS)
    colnames(mat.has) = Sigma
    rownames(mat.has) = c(1:nf)

    # Matrix Spesi
    Spesi = matrix(0, nrow = nf, ncol = nS)
    colnames(Spesi) = Sigma
    rownames(Spesi) = c(1:nf)

    # Matrix Sensi
    Sensi = matrix(0, nrow = nf, ncol = nS)
    colnames(Sensi) = Sigma
    rownames(Sensi) = c(1:nf)

    # Matrix Gmean
    Gmean = matrix(0, nrow = nf, ncol = nS)
    colnames(Gmean) = Sigma
    rownames(Gmean) = c(1:nf)

    # Matrix AUC
    AUCx = matrix(0, nrow = nf, ncol = nS)
    colnames(AUCx) = Sigma
    rownames(AUCx) = c(1:nf)

    # Matrix CV
    CV = matrix(0, nrow = 1, ncol = nC)
    colnames(CV) = Sigma
    #n = 0
    conmat = vector("list", nS)
    for (k in 1:nS) {
      CVx = 0
      Mconmat = matrix(0,nrow = nf, ncol = 4)
      colnames(Mconmat) = c("TN", "FN", "FP", "TP")
      #n = n+1
      for (j in 1:nf) {
        xj = x[-fold_sample[[j]],]
        yj = y[-fold_sample[[j]]]

        hasil                                                    =
try(svm(xj,yj,class.weights=Si(yj),cost=C[i],gamma=Sigma[k]))
        if(inherits(hasil, "try-error")) {
          mat.has[j,k] = 0
        } else {

          Xj = x[fold_sample[[j]],]
          Yj = y[fold_sample[[j]]]
          fx=predict(hasil,Xj)
          akurasi=pred(fx,Yj)
```

```
          conmatx  = akurasi$conmat
          for (con in 1:length(conmatx)) {
            Mconmat[j,con] = conmatx[con]
          }
          mat.has[j,k] = akurasi$accuracy
          Spesi[j,k]   = akurasi$Spesi
          Sensi[j,k]   = akurasi$Sensi
          Gmean[j,k]   = akurasi$Gmean
          AUCx[j,k]    = akurasi$AUC

        }
      }
      for (j in 1:nf) {
        CVx           =           CVx           +           (1-
mat.has[j,k])*length(fold_sample[[j]])/length(y)
      }
      CV[,i] = CVx
      conmat[[k]] = Mconmat
    }
    result[[i]] = list(Akurasi = mat.has,
                       Sensitivity = Sensi,
                       Specificity = Spesi,
                       Gmeans = Gmean,
                       AUC = AUCx)
  }
  names(result) = C
  return(result)
}
```

Lampiran 3. Syntax Klasifikasi dengan EFSVM

```
quad<-function(x,y,cost, Sigma){
  library(kernlab)
  x = as.matrix(x)
  y = as.matrix(y)
  m=dim(x)[1]
  rbf <- rbfdot(sigma = Sigma)
  ## create H matrix etc.
  H <- kernelPol(rbf,x,,y)
  c <- matrix(rep(-1,m))
  A <- t(y)
  b <- 0
  l <- matrix(rep(0,m))
  u <- cost
  r <- 0

  capture.output(sv  <-  ipop(c,H,A,b,l,u,r,verb=TRUE,  sigf=5,
margin=1e-8))
  ipopsol<-primal(sv)
  alpha<-matrix(ipopsol, nrow=m)

  #--------------------------------------------------#
  # Calculation of the normal vector W and bias term b
  #--------------------------------------------------#
  w=t(alpha*y)%*%(x) #W
  ff=as.matrix(matrix(rep(alpha*y,m),m,m))%*%H
  fout=matrix(t(apply(ff,2,sum)))
  pos=which(alpha>1e-6)
  b = mean(y[pos]-fout[pos]) #b
  list(W = w, b = b)
}

# grid search
gridSearchx = function(y, x, fun, C, Sigma, fold_sample) {
  #------------------#
  #   EbasedFuz
  #------------------#

EbasedF = function(x, y, Beta, kNN, m) {
  y = as.numeric(y)
  ndata = dim(x)[1]
  rownames(x) = c(1:ndata)
  jarak1 = as.matrix(dist(x, method = "euclidean", diag = TRUE,
upper = TRUE, p = 2))
  coba=1000*diag(ndata)
  jarak=coba+jarak1
  # H
  H = rep(0, ndata)
  LN = data.frame()
  for (i in 1:ndata){
    Nterdekat = as.numeric(names(sort(jarak[i,]))[1:kNN])
    # L=label
    LNi = y[Nterdekat]
    LN = rbind(LN,LNi)
    Hpi = table(LNi)/kNN
    lHpi = length(as.vector(Hpi))
    if (lHpi>1) {
```

```
      H[i] = -Hpi[2]*log(Hpi[2])-Hpi[1]*log(Hpi[1])
    } else {
      H[i] = 0
    }
  }
  colnames(LN) = paste0("NN",c(1:kNN))
  # Subl
  Sub = vector("list", m)
  thr = matrix(0, m, 2)
  for (l in 1:m) {
    thrUp = min(H) + (l/m)*(max(H) - min(H))
    thrLow = min(H) + ((l-1)/m)*(max(H) - min(H))
    for (i in 1:ndata) {
      if (y[i]==-1) {
        if (H[i]>=thrLow & H[i]<=thrUp) {
          Sub[[l]] = c(Sub[[l]], i)
        }
      }
    }
    thr[l,1] = thrLow
    thr[l,2] = thrUp
  }
  # FM
  FM = c()
  for (l in 1:m){
    Fmx = 1-Beta*(l-1)
    FM = c(FM, Fmx)
  }
  # si
  Si = rep(1, ndata)
  for (l in 1:m) {
    Subl = Sub[[l]]
    Si[Subl] = FM[l]
  }
  list(dist = jarak, H = H, thr = thr,
       Subl = Sub, FM = FM, si = Si, LN = LN)
}

  #------------------#
  #    ACCURACY
  #------------------#

  #y = dataxy
  #x = dataxx
  #fun = quad
  #C = barisC
  #Sigma = barisSigma
  #fold_sample = fold
  pred<-function(x,lable){
    C=NULL
    n = length(x)
    for (i in 1:n){
      if(x[i]>0){C[i]=1}
      else {C[i]=-1}
    }

    TP = 0
    for (i in 1:n) {if (C[i]==1 & lable[i]==1) (TP=TP+1)}
```

```
  FN = 0
  for (i in 1:n) {if (C[i]!=1 & lable[i]==1) (FN=FN+1)}

  FP = 0
  for (i in 1:n) {if (C[i]==1 & lable[i]!=1) (FP=FP+1)}

  TN = 0
  for (i in 1:n) {if (C[i]!=1 & lable[i]!=1) (TN=TN+1)}

  Sensi = TP/(TP+FN)
  Spesi = TN/(TN+FP)
  Gmean = sqrt(Sensi*Spesi)
  AUC   = (1+(TP/(TP+FN))-(FP/(FP+TN)))/2

  accuracy=mean(C==lable)
  result=list(C,accuracy)
  conmat=table(C,lable) #confusion matrix
  list(accuracy=accuracy, conmat=conmat,
       Sensi = Sensi, Spesi = Spesi,
       Gmean = Gmean, AUC = AUC)
}

y           = as.numeric(as.matrix(y))
x           = x
C           = C
Sigma       = Sigma
fold_sample = fold_sample
nf          = length(fold_sample)
nC          = length(C)
nS          = length(Sigma)


result = vector("list", nC)

for (i in 1:nC){
  # Matrix Akurasi
  mat.has = matrix(0, nrow = nf, ncol = nS)
  colnames(mat.has) = Sigma
  rownames(mat.has) = c(1:nf)

  # Matrix Spesi
  Spesi = matrix(0, nrow = nf, ncol = nS)
  colnames(Spesi) = Sigma
  rownames(Spesi) = c(1:nf)

  # Matrix Sensi
  Sensi = matrix(0, nrow = nf, ncol = nS)
  colnames(Sensi) = Sigma
  rownames(Sensi) = c(1:nf)

  # Matrix Gmean
  Gmean = matrix(0, nrow = nf, ncol = nS)
  colnames(Gmean) = Sigma
  rownames(Gmean) = c(1:nf)

  # Matrix AUC
  AUCx = matrix(0, nrow = nf, ncol = nS)
  colnames(AUCx) = Sigma
  rownames(AUCx) = c(1:nf)
```

```
    # Matrix CV
    CV = matrix(0, nrow = 1, ncol = nC)
    colnames(CV) = Sigma
    #n = 0
    conmat = vector("list", nS)
    for (k in 1:nS) {
      CVx = 0
      Mconmat = matrix(0,nrow = nf, ncol = 4)
      colnames(Mconmat) = c("TN", "FN", "FP", "TP")
      #n = n+1
      for (j in 1:nf) {
        xj = x[-fold_sample[[j]],]
        yj = y[-fold_sample[[j]]]

        Si = EbasedF(x = xj,y = yj,Beta = 0.05,kNN = nf, m = 7)
        si = Si$si
        hasil = try(fun(xj,yj,C[i]*si,Sigma[k]))

        if(inherits(hasil, "try-error")) {
          mat.has[j,k] = 0
        } else {

          Xj = x[fold_sample[[j]],]
          Yj = y[fold_sample[[j]]]
          fx=t(hasil$W %*% t(as.matrix(Xj))) + hasil$b
          akurasi=pred(fx,Yj)
          conmatx  = akurasi$conmat
          for (con in 1:length(conmatx)) {
            Mconmat[j,con] = conmatx[con]
          }
          mat.has[j,k] = akurasi$accuracy
          Spesi[j,k]   = akurasi$Spesi
          Sensi[j,k]   = akurasi$Sensi
          Gmean[j,k]   = akurasi$Gmean
          AUCx[j,k]    = akurasi$AUC

        }
      }
      for (j in 1:nf) {
        CVx           =           CVx           +           (1-
mat.has[j,k])*length(fold_sample[[j]])/length(y)
      }
      CV[,i] = CVx
      conmat[[k]] = Mconmat
    }
    result[[i]] = list(#Akurasi = mat.has,
                       #Sensitivity = Sensi,
                       #Specificity = Spesi,
                       #Gmeans = Gmean,
                       AUC = AUCx)
  }
  names(result) = C
  return(result)
}
```

Lampiran 4. Syntax FCBF

```
#Feature Selection dengan FCBF
library(gtools)
library(Biocomb)
library(Rcpp)
data2[,ncol(data2)]   =   as.factor(data2[,ncol(data2)])   #data
breast cancer
data4[,ncol(data4)]   =   as.factor(data4[,ncol(data4)])   #data
colon cancer
attrs.nominal=numeric()
system.time(select.fast.filter(data1,disc.method="MDL",
threshold=0,attrs.nominal=attrs.nominal))
system.time(select.fast.filter(data2,disc.method="MDL",
threshold=0,attrs.nominal=attrs.nominal))
system.time(select.fast.filter(data4,disc.method="MDL",
threshold=0.1,attrs.nominal=attrs.nominal))
out=select.fast.filter(data1,disc.method="MDL",
threshold=0,attrs.nominal=attrs.nominal)
out2=select.fast.filter(data2,disc.method="MDL",
threshold=0,attrs.nominal=attrs.nominal)
out4=select.fast.filter(data4,disc.method="MDL",
threshold=0.1,attrs.nominal=attrs.nominal)
```

Lampiran 5. AUC dari Klasifikasi Data *Breast Cancer* dengan SVM

Seluruh Feature								
γ\C	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8
2^-18	50.00	50.00	49.57	54.37	61.03	67.94	70.59	72.26
2^-17	50.00	49.57	54.81	61.03	67.94	70.59	72.26	73.37
2^-16	49.57	55.64	60.16	67.94	70.59	71.82	73.37	73.37
2^-15	56.75	61.00	63.19	67.45	69.07	70.78	70.78	70.78
2^-14	61.83	68.06	70.12	70.55	72.93	72.93	72.93	72.93
2^-13	66.13	71.75	71.75	71.75	71.75	71.75	71.75	71.75
2^-12	68.47	70.14	70.14	70.14	70.14	70.14	70.14	70.14
2^-11	53.59	53.59	53.59	53.59	53.59	53.59	53.59	53.59
2^-10	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-9	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-8	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-7	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-6	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-5	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-4	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
Selected Feature								
2^-18	98.06	98.06	98.06	98.06	98.06	98.06	98.06	98.06
2^-17	98.06	98.06	98.06	98.06	98.06	98.06	78.06	98.06
2^-16	98.06	98.06	98.06	98.06	98.06	98.06	78.06	98.06
2^-15	98.06	98.06	98.06	98.06	98.06	78.06	78.06	98.06
2^-14	98.06	98.06	98.06	98.06	78.06	98.06	98.06	98.89
2^-13	98.06	98.06	98.06	98.06	98.06	98.06	78.89	100.00
2^-12	98.06	98.06	98.06	98.06	98.06	98.89	100.00	99.17
2^-11	98.06	98.06	98.06	78.06	58.89	80.00	60.00	80.00
2^-10	98.06	98.06	98.06	60.00	80.00	80.00	99.57	99.57
2^-9	96.75	96.75	98.02	79.13	79.13	60.00	97.83	97.83
2^-8	92.26	93.52	96.77	96.43	96.56	96.56	96.56	96.56
2^-7	90.70	93.55	93.77	94.20	75.16	94.29	94.29	94.29
2^-6	89.65	91.41	91.84	91.41	90.97	90.97	90.97	90.97
2^-5	90.88	90.79	90.36	90.36	90.36	90.36	90.36	90.36
2^-4	91.19	90.32	90.32	90.32	89.89	89.89	89.89	90.32

Lampiran 6. AUC dari Klasifikasi Data Colon Cancer dengan SVM

Seluruh Feature								
γ\C	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8
2^-18	50	50	50	50	65.75	75.25	78	80
2^-17	50	50	50	65.75	75.25	76	80	78
2^-16	50	50	65.75	75.25	76	80	78	79.25
2^-15	50	63.75	75.25	76	80	79.25	86.25	86.25
2^-14	57.75	75.25	77.25	80	84.25	84.25	84.25	84.25
2^-13	73.25	75.25	86.25	84.25	83	83	83	83
2^-12	76.5	86.25	84.25	84.25	84.25	84.25	84.25	84.25
2^-11	79.5	80.25	80.25	80.25	80.25	80.25	80.25	80.25
2^-10	75.5	77.5	77.5	77.5	77.5	77.5	77.5	77.5
2^-9	56.5	56.5	56.5	56.5	56.5	56.5	56.5	56.5
2^-8	50	50	50	50	50	50	50	50
2^-7	50	50	50	50	50	50	50	50
2^-6	50	50	50	50	50	50	50	50
2^-5	50	50	50	50	50	50	50	50
2^-4	50	50	50	50	50	50	50	50
Selected Feature								
2^-18	50	50	50	50	50	50	50	50
2^-17	50	50	50	50	50	50	50	50
2^-16	50	50	50	50	50	50	50	69
2^-15	50	50	50	50	50	50	69	83.75
2^-14	50	50	50	50	50	69	83.75	84.5
2^-13	50	50	50	50	69	83.75	84.5	83.25
2^-12	50	50	50	69	83.75	84.5	83.25	84.5
2^-11	50	50	69	83.75	84.5	83.25	84.5	84.5
2^-10	50	69	83.75	84.5	83.25	83.25	83.25	82
2^-9	69	82.5	83.25	83.25	83.25	83.25	80.75	82
2^-8	82.5	83.25	83.25	88.25	82	82	82	83.5
2^-7	83.25	85.25	83.25	84	82	86	77	73.75
2^-6	85.25	85.25	82	84	78.25	75	70.5	71.75
2^-5	85.25	80.75	84.75	78.25	76.25	77.5	77.5	77.5
2^-4	80.75	82.75	80.75	78.75	78.75	78.75	78.75	78.75

Lampiran 7. AUC dari Klasifikasi Data *Breast Cancer* dengan FSVM

Seluruh Feature								
γ\C	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8
2^-18	50.00	50.00	50.00	50.00	50.00	66.43	96.96	99.57
2^-17	50.00	50.00	50.00	50.43	66.43	93.48	98.70	98.70
2^-16	50.00	50.00	50.43	66.43	86.96	95.22	95.22	95.22
2^-15	50.00	51.30	64.07	77.55	84.07	84.07	84.07	84.07
2^-14	51.30	63.02	71.90	75.81	75.81	75.81	75.81	75.81
2^-13	54.08	64.43	66.25	66.25	66.25	66.25	66.25	66.25
2^-12	53.54	53.54	53.54	53.54	53.54	53.54	53.54	53.54
2^-11	49.35	50.22	50.22	50.22	50.22	50.22	50.22	50.22
2^-10	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-9	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-8	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-7	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-6	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-5	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
2^-4	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
Selected Feature								
2^-18	98.06	98.06	98.06	98.06	98.06	98.06	98.06	98.06
2^-17	98.06	98.06	98.06	98.06	98.06	98.06	78.06	98.06
2^-16	98.06	98.06	98.06	98.06	98.06	98.06	78.06	98.06
2^-15	98.06	98.06	98.06	98.06	98.06	78.06	78.06	98.06
2^-14	98.06	98.06	98.06	98.06	78.06	98.06	98.06	98.89
2^-13	98.06	98.06	98.06	98.06	98.06	98.06	78.89	100.00
2^-12	98.06	98.06	98.06	98.06	98.06	98.89	100.00	99.17
2^-11	98.06	98.06	98.06	78.06	58.89	80.00	60.00	80.00
2^-10	98.06	98.06	98.06	60.00	80.00	80.00	99.57	99.57
2^-9	96.75	96.75	98.02	79.13	79.13	60.00	97.83	97.83
2^-8	92.26	93.52	96.77	96.43	96.56	96.56	96.56	96.56
2^-7	90.70	93.55	93.77	94.20	75.16	94.29	94.29	94.29
2^-6	89.65	91.41	91.84	91.41	90.97	90.97	90.97	90.97
2^-5	90.88	90.79	90.36	90.36	90.36	90.36	90.36	90.36
2^-4	91.19	90.32	90.32	90.32	89.89	89.89	89.89	90.32

Lampiran 8. AUC dari Klasifikasi Data Colon Cancer dengan FSVM

Seluruh Feature								
γ\C	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8
2^-18	50	50	50	50	50	57.5	78.75	80
2^-17	50	50	50	50	57.5	76.25	80	83.75
2^-16	50	50	50	58.75	73.75	78.75	81.25	81.25
2^-15	50	50	60	73.75	76.25	78.75	78.75	78.75
2^-14	50	58.75	72.5	75	80	80	80	80
2^-13	52.5	67.5	72.5	73.75	73.75	73.75	73.75	73.75
2^-12	62.5	68.75	72.5	72.5	72.5	72.5	72.5	72.5
2^-11	60	65	66.25	66.25	66.25	66.25	66.25	66.25
2^-10	52.5	52.5	52.5	52.5	52.5	52.5	52.5	52.5
2^-9	50	50	50	50	50	50	50	50
2^-8	50	50	50	50	50	50	50	50
2^-7	50	50	50	50	50	50	50	50
2^-6	50	50	50	50	50	50	50	50
2^-5	50	50	50	50	50	50	50	50
Selected Feature								
2^-18	50	50	50	50	50	50	50	50
2^-17	50	50	50	50	50	50	50	50
2^-16	50	50	50	50	50	50	50	50
2^-15	50	50	50	50	50	50	50	52.5
2^-14	50	50	50	50	50	50	52.5	70
2^-13	50	50	50	50	50	52.5	70	83.75
2^-12	50	50	50	50	52.5	70	82.5	91.25
2^-11	50	50	50	52.5	70	82.5	91.25	95
2^-10	50	50	51.25	71.25	81.25	91.25	91.25	88.75
2^-9	50	51.25	68.75	80	88.75	87.5	86.25	85
2^-8	50	68.75	75	85	87.5	86.25	86.25	86.25
2^-7	66.25	71.25	78.75	83.75	86.25	82.5	83.75	83.75
2^-6	70	76.25	80	81.25	77.5	77.5	77.5	77.5
2^-5	72.5	73.75	73.75	73.75	73.75	73.75	73.75	73.75
2^-4	71.25	71.25	71.25	71.25	71.25	71.25	71.25	71.25

Lampiran 9. AUC dari Klasifikasi Data *Breast Cancer* dengan EFSVM

Seluruh Feature								
γ\C	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8
2^-18	97.46	97.46	97.46	97.46	98.73	80.00	80.00	98.26
2^-17	97.46	97.46	97.46	77.46	80.00	100.00	100.00	98.26
2^-16	97.46	97.46	97.46	79.57	98.26	100.00	100.00	98.26
2^-15	97.46	97.03	80.00	99.57	98.26	98.26	98.26	98.26
2^-14	97.03	99.13	99.57	98.26	98.26	98.26	98.26	98.26
2^-13	97.43	98.26	98.26	98.26	98.26	98.26	98.26	98.26
2^-12	92.53	94.23	94.23	94.23	94.23	94.23	94.23	94.23
2^-11	92.06	92.49	92.49	92.49	92.49	92.49	92.49	92.49
2^-10	89.92	89.92	89.92	89.92	89.92	89.92	89.92	89.92
2^-9	88.65	88.65	88.65	88.65	88.65	88.65	88.65	88.65
2^-8	89.61	89.61	89.61	89.61	89.61	89.61	89.61	89.61
2^-7	90.14	90.14	90.14	90.14	90.14	90.14	90.14	90.14
2^-6	90.14	90.14	90.14	90.14	90.14	90.14	90.14	90.14
2^-5	90.14	90.14	90.14	90.14	90.14	90.14	90.14	90.14
2^-4	90.14	90.14	90.14	90.14	90.14	90.14	90.14	90.14
Selected Feature								
2^-18	95.23	95.23	76.10	77.39	77.39	95.23	95.23	95.23
2^-17	95.23	95.23	75.23	95.23	95.23	95.23	95.23	95.23
2^-16	95.23	95.23	95.23	95.23	95.23	95.23	95.23	76.10
2^-15	95.23	95.23	95.23	95.23	95.23	95.23	95.23	95.23
2^-14	95.23	95.23	95.23	95.23	95.23	95.23	95.23	96.86
2^-13	95.23	95.23	95.23	95.23	95.23	95.23	96.86	99.13
2^-12	95.23	95.23	95.23	95.23	95.23	77.30	99.13	99.57
2^-11	94.79	94.79	94.79	95.23	96.86	98.70	99.57	100.00
2^-10	75.66	94.36	95.23	78.17	98.26	98.70	99.57	99.57
2^-9	94.36	94.79	77.83	97.83	97.83	97.83	97.83	97.83
2^-8	95.47	96.34	76.43	96.56	96.56	96.56	96.56	96.56
2^-7	94.51	93.77	94.73	94.29	94.29	94.29	94.29	94.29
2^-6	91.32	91.41	90.97	90.97	90.97	90.97	90.97	90.97
2^-5	90.36	90.36	90.36	90.36	90.36	90.36	90.36	90.36
2^-4	89.89	89.89	89.89	89.89	89.89	90.32	89.89	89.89

Lampiran 10. AUC dari Klasifikasi Data Colon Cancer dengan EFSVM

Seluruh Feature								
γ\C	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8
2^-18	91.75	91.75	91.75	73.75	91.75	91.75	93.75	60
2^-17	91.75	91.75	91.75	91.75	91.75	93.75	75	93.75
2^-16	91.75	91.75	91.75	91.75	77.5	75	76.25	96.25
2^-15	90.5	91.75	71.75	77.5	77.5	93.75	96.25	96.25
2^-14	90.5	90.5	76.25	57.5	95	95	95	95
2^-13	89.25	89.25	91.25	93.75	93.75	93.75	93.75	93.75
2^-12	89.25	72.5	91.25	91.25	91.25	91.25	91.25	91.25
2^-11	85.5	86.25	85	85	85	85	85	85
2^-10	80.5	79.25	79.25	79.25	79.25	79.25	79.25	79.25
2^-9	78	78	78	78	78	78	78	78
2^-8	80.5	80.5	80.5	80.5	80.5	80.5	80.5	80.5
2^-7	78.5	78.5	78.5	78.5	78.5	78.5	78.5	78.5
2^-6	78.5	78.5	78.5	78.5	78.5	78.5	78.5	78.5
2^-5	78.5	78.5	78.5	78.5	78.5	78.5	78.5	78.5
2^-4	78.5	78.5	78.5	78.5	78.5	78.5	78.5	78.5
Selected Feature								
2^-18	38.75	58.75	78.75	94.25	94.25	94.25	94.25	74.25
2^-17	58.75	58.75	94.25	94.25	94.25	94.25	94.25	74.25
2^-16	58.75	74.25	94.25	94.25	94.25	94.25	94.25	94.25
2^-15	74.25	94.25	78.75	94.25	94.25	94.25	94.25	75.5
2^-14	94.25	94.25	94.25	94.25	94.25	94.25	75.5	77.5
2^-13	94.25	94.25	94.25	94.25	94.25	94.25	96.25	55.5
2^-12	94.25	94.25	94.25	94.25	94.25	96.25	38.75	58.75
2^-11	94.25	94.25	94.25	74.25	58.75	95.5	58.75	77.5
2^-10	94.25	94.25	74.25	56.25	95.5	76.25	77.5	53.75
2^-9	94.25	94.25	56.25	75.5	56.25	57.5	73.75	73.75
2^-8	94.25	74.25	75.5	57.5	73.75	92.5	92.5	92.5
2^-7	74.25	95.5	75.5	92.5	92.5	92.5	92.5	92.5
2^-6	94.25	75.5	92.5	92.5	92.5	92.5	92.5	92.5
2^-5	91.75	92.5	92.5	92.5	92.5	92.5	92.5	92.5
2^-4	90	90	90	90	90	90	90	90

Lampiran 11. Hasil α dan b untuk Data Breast Cancer

	Seluruh Feature	Selected Feature		Seluruh Feature	Selected Feature
	α	α		α	α
[1,]	3.243244	0.997539	[26,]	2.581021	1.01679
[2,]	5.38E-15	0.294178	[27,]	7.352563	1.074179
[3,]	3.045192	0.151651	[28,]	3.986016	1.020889
[4,]	1.248342	0.815881	[29,]	0.866138	0.891253
[5,]	5.858075	0.724486	[30,]	4.801693	0.181889
[6,]	1.88E-19	0.643224	[31,]	4.164965	0.851589
[7,]	4.5491	0.170251	[32,]	1.007526	0.986781
[8,]	1.98E-19	0.921406	[33,]	1.770976	2.43E-12
[9,]	1.670354	0.919266	[34,]	1.32E-20	0.404814
[10,]	1.042287	0.734552	[35,]	1.557269	0.98939
[11,]	0.425847	0.913554	[36,]	3.973056	0.953934
[12,]	0.480149	0.98223	[37,]	2.553	0.637857
[13,]	0.411045	0.830023	[38,]	1.141145	0.280103
[14,]	0.836879	0.670473	[39,]	6.248602	0.936578
[15,]	1.22E-20	0.793463	[40,]	1.3545	1.81E-12
[16,]	2.878093	1.022006	[41,]	0.714514	0.964009
[17,]	0.584169	0.352311	[42,]	9.033124	0.968418
[18,]	7.136214	1.064202	[43,]	15.47272	1.071618
[19,]	3.62E-20	6.51E-12	[44,]	0.041077	0.977989
[20,]	1.97E-16	0.959255	[45,]	5.761008	1.0173
[21,]	0.329758	0.863184	[46,]	0.13451	0.967988
[22,]	4.060288	0.691873	[47,]	1.005237	0.978017
[23,]	2.262149	0.934598	[48,]	4.145632	0.979725
[24,]	8.94E-18	0.239957	[49,]	2.199247	0.977166
[25,]	3.486568	1.003506	[50,]	12.68183	1.461938

	Seluruh Feature	Selected Feature		Seluruh Feature	Selected Feature
	α	α		α	α
[51,]	3.235936	0.977864	[76,]	2.685656	2.12E-09
[52,]	5.279916	0.850309	[77,]	7.09E-21	8E-14
[53,]	2.313267	0.977595	[78,]	5.871366	1.001463
[54,]	7.783933	1.190968	[79,]	7.612291	0.727263
[55,]	0.650432	0.977733	[80,]	11.39935	1.102111
[56,]	1.550467	0.977975	[81,]	8.781731	1.256406
[57,]	3.556949	0.934333	[82,]	1.337698	1.45E-11
[58,]	2.796232	0.979422	[83,]	3.309735	0.413683
[59,]	1.201271	0.971009	[84,]	11.63432	0.612651
[60,]	6.533136	0.967445	[85,]	12.29427	0.861887
[61,]	8.759205	0.575275	[86,]	10.38651	1.042695

[62,]	3.93E-20	0.502708	[87,]	11.79819	1.183895
[63,]	9.91E-20	0.747008	[88,]	7.321783	0.838681
[64,]	4.69E-20	1.21E-16	[89,]	9.861695	1.13291
[65,]	1.43E-17	0.704855	[90,]	1.51E-20	0.897011
[66,]	5.686973	0.244513	[91,]	4.427678	0.978694
[67,]	1.138145	0.29943	[92,]	16	1.311865
[68,]	4.742856	0.302726	[93,]	2.404628	0.968037
[69,]	6.757037	0.913906	[94,]	16	2.126928
[70,]	2.232502	1.08E-13	[95,]	14.27727	0.990761
[71,]	4.3228	0.549206	[96,]	3.995819	0.931027
[72,]	1.95E-19	8.15E-15	[97,]	4.143213	0.99043
[73,]	3.975697	0.731375	[98,]	4.953062	0.324381
[74,]	0.909815	0.035673	[99,]	6.47E-21	1.59E-10
[75,]	1.34E-20	0.107019	[100,]	11.64851	1.22E-11

	Seluruh Feature	Selected Feature		Seluruh Feature	Selected Feature
	α	α		α	α
[101,]	10.71045	0.566034	[124,]	8.293303	0.910516
[102,]	9.725245	1.41E-13	[125,]	16	1.903772
[103,]	7.17E-15	3.49E-13	[126,]	3.62E-20	0.9665
[104,]	0.39456	0.849147	[127,]	10.99953	1.341157
[105,]	8.51E-20	1.78E-13	[128,]	5.946153	1.162148
[106,]	6.67E-20	7.37E-09	[129,]	1.02E-19	0.976754
[107,]	4.523468	0.34371	[130,]	2.436163	0.978234
[108,]	6.16E-20	0.802975	[131,]	15.7317	1.121913
[109,]	3.018155	1.021733	[132,]	9.556046	0.98502
[110,]	6.57003	0.479821	[133,]	1.427811	0.97294
[111,]	8.336075	0.770512	[134,]	2.91E-21	0.757737
[112,]	2.18E-20	0.216323		b	b
[113,]	7.147749	1.060444		-0.23364	-0.21368
[114,]	2.857416	2.02E-11			
[115,]	6.872806	0.266527			
[116,]	7.530682	0.070785			
[117,]	4.85E-21	5.28E-12			
[118,]	16	1.054513			
[119,]	6.33E-21	1.79E-14			
[120,]	2.957919	1.010992			
[121,]	0.044279	1.011565			
[122,]	3.058342	0.120832			
[123,]	15.01992	1.244599			

Lampiran 12. Hasil α dan b untuk Data Colon Cancer

	Seluruh Feature	Selected Feature
	α	α
[1,]	18.84174	1.6
[2,]	49.27742	1.6
[3,]	69.78453	2
[4,]	44.40941	2
[5,]	37.78651	2
[6,]	8.882415	1.6
[7,]	12.0748	1.31169E-12
[8,]	6.33098E-18	1.25418E-13
[9,]	9.48958	1.007967
[10,]	7.71936E-18	4.01953E-08
[11,]	8.910999	2
[12,]	24.70505	1.6
[13,]	7.618393	2
[14,]	28.49474	1.6
[15,]	35.02853	2
[16,]	5.821127	2
[17,]	30.23264	2
[18,]	26.62338	2
[19,]	42.24011	1.44162E-12
[20,]	79.9143	2
[21,]	3.88272E-19	2
[22,]	2.71287E-17	4.19007E-14
[23,]	20.47275	2
[24,]	1.85634E-18	2.79911E-14
[25,]	1.75499E-18	2.38118E-13
[26,]	4.03704E-18	1.42561E-12
[27,]	10.76177	1.7
[28,]	7.15784E-16	2
[29,]	4.0055E-17	4.65423E-14
[30,]	8.68386E-18	1.7
[31,]	6.68924E-17	5.08039E-14
[32,]	27.28005	2
[33,]	6.56984E-18	4.54832E-13
[34,]	50.89135	1.6
[35,]	14.50297	2
[36,]	4.801721	-2.09159E-19
[37,]	1.39702E-17	1.45243E-12
[38,]	24.74874	1.7
[39,]	1.27777E-17	8.13817E-14

[40,]	1.33613E-16	1.10306E-11
[41,]	21.0721	2
[42,]	80	1.7
[43,]	12.39847	2
[44,]	7.18103E-18	2.69333E-12
[45,]	2.306501	2
[46,]	73.25214	2
[47,]	19.96221	1.6
[48,]	1.20543E-17	1.307967
[49,]	14.03287	1.7
[50,]	19.28711	2
	b	b
	0.09090909	-0.05882353

(Halaman ini sengaja dikosongkan)

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini, mahasiswa Departemen Statistika FMKSD ITS:

Nama : Faroh Ladayya

NRP : 06211550010202

Program Studi : Magister Statistika

menyatakan bahwa data yang digunakan dalam ini merupakan data sekunder yang diambil dari website yaitu:

Sumber : *http://datam.i2r.a-star.edu.sg/datasets/krbd*

Keterangan : -

Surat pernyataan ini dibuat dengan sebenarnya. Apabila terdapat pemalsuan data maka saya siap menerima sanksi sesuai aturan yang berlaku.

Surabaya, 23 Januari 2018

Mengetahui

Pembimbing Tesis	Mahasiswa
(Santi Wulan Purnami, M.Si, Ph. D)	(Faroh Ladayya)
NIP. 19720923 199803 2 001	NRP. 06211550010202

BIODATA PENULIS

Penulis memiliki nama lengkap Faroh Ladayya. Penulis lahir di Tulungagung, pada tanggal 28 Januari tahun 1994. Jenjang pendidikan yang telah ditempuh penulisa dalah RA Halimah Asa'diah padatahun 1998-1999. Menempuh Sekolah Dasar di MI Manba'ul Ulum pada(1999-2005), SMP Negeri 1 Ngunut (2005-2008), SMA Negeri 1 Blitar (2008-2011). Pendidikan tinggi dimulai pada tahun 2011 di Jurusan Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA), Institut Teknologi Sepuluh Nopember, Surabaya dan menyelesaikan program S-1 pada tahun 2015. Kemudian pada tahun 2015 dengan bantuan beasiswa LPDP, penulis melanjutkan program pascasarjana S-2 di Institut Teknologi Sepuluh Nopember (ITS), Jurusan Statistika, Fakultas Matematika, Komputasi, dan Sains Data (FMKSD). Jika terdapat kritik dan saran mengenai tesis yang penulis buat ini dapat menghubungi penulis melalui *E-mail* di *ladayyafaroh@gmail.com*.